

**APLIKASI FUZZY LOGIC CONTROL
PADA ALAT QUALITY CONTROL AIR MINUM
PDAM YANG DIINTERFACEKAN KE KOMPUTER**

TUGAS AKHIR

Oleh :

YUSUF LESTANTO

2290.100.044

RSE

629.831 2

Les

a-1

1996



**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
1996**

| PERPUSTAKAAN I T S | |
|-----------------------|-------------|
| Tgl. Terima | 09 JAN 1997 |
| Terima Dari | H |
| No. Agenda | 6795 |

**APLIKASI FUZZY LOGIC CONTROL
PADA ALAT QUALITY CONTROL AIR MINUM
PDAM YANG DIINTERFACEKAN KE KOMPUTER**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik Elektro**

Pada

Bidang Studi Elektronika

Jurusan Teknik Elektro

Fakultas Teknologi Industri


Institut Teknologi Sepuluh Nopember

S u r a b a y a

Mengetahui / Menyetujui

Dosen Pembimbing I

Dosen Pembimbing II



Ir. H. MOCH. HEROE



RACHMAD SETIAWAN, S.T.

S U R A B A Y A

Nopember, 1996

ABSTRAK

Dalam tugas akhir ini direncanakan suatu prototipe alat quality control air minum dengan menggunakan logika fuzzy yang diinterfacekan ke komputer. Parameter yang diamati adalah derajat keasam-basaan (pH), kadar oksigen terlarut (DO), dan tingkat kejernihan. Dengan menggunakan sensor pH, DO, dan kejernihan parameter-parameter tersebut diubah ke tegangan yang kemudian dibaca ADC. Hasil pembacaan ini selanjutnya diolah dengan menggunakan logika fuzzy untuk penambahan larutan pengkondisi, sehingga akan didapat air sesuai dengan persyaratan kesehatan.

Dengan alat ini dapat dilakukan pengecekan tiga parameter air, sehingga keadaan air dapat langsung dimonitor. Setelah proses pengkondisian dapat ditentukan apakah air baku dapat diolah atau pun air minum telah siap didistribusikan ke pelanggan.

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Allah SWT atas berkat rahmat dan kuasanya, sehingga dapat menyelesaikan Tugas Akhir yang berjudul :

APLIKASI FUZZY LOGIC CONTROL PADA ALAT QUALITY CONTROL AIR MINUM PDAM YANG DIINTERFACEKAN KE KOMPUTER

Tugas Akhir ini merupakan salah satu syarat dalam menyelesaikan pendidikan di Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember Surabaya.

Tugas Akhir ini dibuat berdasarkan teori-teori yang diperoleh selama masa perkuliahan dan berbagai buku literatur penunjangnya.

Dengan selesainya Tugas Akhir ini, penyusun menyampaikan terima kasih kepada :

1. Bapak Ir. H. Moch Heroe, selaku Dosen Pembimbing I, atas segala bimbingan dan arahannya.
2. Bapak Rachmad Setiawan ST, selaku Dosen Pembimbing II.
3. Bapak Ir. Soetikno, selaku Koordinator Bidang Studi Elektronika Jurusan Teknik Elektro FTI-ITS.
4. Bapak DR. Ir. Moch. Salehudin, M. Eng. Sc., selaku Ketua Jurusan Teknik Elektro FTI-ITS.
5. Bapak Ir. Karyadi, MSc., selaku Dosen Wali.
6. Seluruh Staf Dosen Bidang Studi Elektronika dan Karyawan di Jurusan Teknik Elektro FTI-ITS.
7. Seluruh Staf dan Karyawan Laboratorium Induk IPA Ngagel atas segala bantuan dan bimbingannya.
8. Kakakku Mbak Yuyun dan Adik-adikku Didik dan Ari atas segala dorongan dan pengertiannya.

9. Eka, Helos, Helmi, Aris, Agus, Dadik, Udank, Mega, Nefy, Lilik, dan teman-teman lainnya.

Penyusun berharap kiranya Tugas Akhir ini dapat bermanfaat bagi yang memerlukannya, khususnya teman-teman di Bidang Studi Elektronika Jurusan Teknik Elektro Fakultas Teknologi Institut Teknologi Sepuluh Nopember Surabaya.

Surabaya, Nopember 1996

Penulis

DAFTAR ISI

| | Halaman |
|---------------------------------|---------|
| JUDUL | i |
| LEMBAR PENGESAHAN | ii |
| ABSTRAK | iii |
| KATA PENGANTAR | iv |
| DAFTAR ISI | vi |
| DAFTAR GAMBAR | xi |
| DAFTAR TABEL | xiii |
| BAB 1 PENDAHULUAN | |
| 1.1 Latar Belakang | 1 |
| 1.2 Permasalahan | 2 |
| 1.3 Batasan Masalah | 2 |
| 1.4 Tujuan | 2 |
| 1.5 Metodologi | 3 |
| 1.6 Sistematika | 3 |
| 1.7 Relevansi | 5 |
| BAB II TEORI PENUNJANG | |
| 2.1 Elektrokimia | 6 |
| 2.1.1 Arus Dan Perubahan Kimia | 6 |
| 2.1.2 Sel Elektrokimia | 7 |
| 2.1.3 Elektroda | 9 |
| 2.1.3.1 Elektroda Calomel (SCE) | 10 |
| 2.1.3.2 Elektroda Gelas | 10 |
| 2.1.3.3 Elektroda Membran | 11 |

| | | |
|---------|--|----|
| 2.1.4 | Pengukuran pH | 12 |
| 2.1.4.1 | Hubungan pH Dan Alkalinitas | 13 |
| 2.1.5 | Pengukuran DO | 14 |
| 2.1.5.1 | Pengukuran Dissolved Oxygen (DO) Dengan Metode Winkler | 14 |
| 2.1.6 | Pengukuran Tingkat Kejernihan | 16 |
| 2.2 | Programmable Peripheral Interface (PPI) 8255 | 17 |
| 2.2.1 | Pengoperasian PPI 8255 | 18 |
| 2.3 | Penguat Operasional | 20 |
| 2.3.1 | Pembanding (Komparator) | 20 |
| 2.3.2 | Penguat Differensial | 21 |
| 2.3.3 | Low Pass Filter | 22 |
| 2.3.4 | Penguat Instrumentasi | 23 |
| 2.4 | Pengubahan Sinyal Analog Ke Digital | 24 |
| 2.4.1 | Multiplexer Analog | 24 |
| 2.4.2 | Konversi Analog Ke Digital | 25 |
| 2.4.2.1 | Successive Approximation ADC | 25 |

BAB III TEORI LOGIKA FUZZY

| | | |
|-------|--|----|
| 3.1 | Pendahuluan | 27 |
| 3.2 | Struktur Dasar Logika Fuzzy | 30 |
| 3.2.1 | Unit Fuzzifikasi | 31 |
| 3.2.2 | Unit Data Dasar Pengambilan Keputusan Fuzzy (Knowledge Base) | 31 |
| 3.2.3 | Unit Defuzzifikasi | 32 |
| 3.3 | Chip Fuzzy NLX230 | 33 |
| 3.3.1 | Pendahuluan | 33 |
| 3.3.2 | Deskripsi Rangkaian Dalam NLX230 | 33 |

| | | |
|---------|---|----|
| 3.3.2.1 | Fuzzifier | 34 |
| 3.3.2.2 | Minimum Comparator | 34 |
| 3.3.2.3 | Maximum Comparator | 34 |
| 3.3.3 | Mode operasi NLX230 | 35 |
| 3.3.3.1 | Mode Fuzzifier | 35 |
| 3.3.3.2 | Mode Input | 35 |
| 3.3.3.3 | Mode Output | 35 |
| 3.3.4 | Register-register internal FMC | 36 |
| 3.3.5 | Deskripsi Pin | 39 |
| BAB IV | PERANCANGAN PERANGKAT KERAS DAN PERANGKAT LUNAK | |
| 4.1 | Permasalahan | 43 |
| 4.2 | Desain Perangkat Keras | 43 |
| 4.2.1 | Modul Fuzzy NLX230 | 45 |
| 4.2.2 | Antar-muka Paralel | 46 |
| 4.2.3 | Analog to Digital Converter | 47 |
| 4.2.4 | Multiplexer Analog | 48 |
| 4.2.5 | Instrumentasi pH | 49 |
| 4.2.6 | Instrumentasi Kadar Oksigen Terlarut | 50 |
| 4.2.7 | Instrumentasi Kejernihan Air | 51 |
| 4.2.8 | Driver Katup | 53 |
| 4.3 | Perencanaan Perangkat Lunak | 53 |
| 4.3.1 | Bitmap dari term dan rule | 55 |
| 4.3.2 | Download dan Verify | 55 |
| 4.3.3 | Running | 58 |

| | | |
|--------------------------------|--|----|
| 4.3.4 | Defuzzifier | 59 |
| 4.3.5 | Pembacaan ADC | 59 |
| BAB V PENGUJIAN DAN PENGUKURAN | | |
| 5.1 | Pengujian Masing-Masing Modul | 60 |
| 5.1.1 | Modul Paralel Interface (PPI card 8255) | 60 |
| 5.1.2 | Modul ADC | 60 |
| 5.1.3 | Modul Fuzzy Card NLX230 | 60 |
| 5.1.4 | Modul Driver katup | 61 |
| 5.1.5 | Modul Instrumentasi pH | 61 |
| 5.1.6 | Modul Instrumentasi DO | 61 |
| 5.1.7 | Modul Instrumentasi Kejernihan | 61 |
| 5.2 | Kalibrasi Modul-Modul Instrumentasi | 61 |
| 5.2.1 | Kalibrasi Modul pH | 62 |
| 5.2.2 | Kalibrasi Modul Kejernihan | 63 |
| 5.2.3 | Kalibrasi Modul DO | 64 |
| 5.3 | Data Percobaan Penetralkan | 65 |
| 5.3.1 | Penetralkan Dari Keadaan Basa | 65 |
| 5.3.2 | Penetralkan Dari Keadaan Asam | 66 |
| 5.4 | Data Penambahan Larutan Tawas atau Kapur Tohor | 68 |
| BAB VI PENUTUP | | |
| 6.1 | Kesimpulan | 69 |
| 6.2 | Saran | 70 |
| DAFTAR PUSTAKA | | 71 |
| LAMPIRAN | | |

- ☐ Rangkaian
- ☐ Bitmap Term dan Rule
- ☐ Listing Program
- ☐ Data Sheet NLX230

RIWAYAT HIDUP

DAFTAR GAMBAR

| | |
|---|----|
| Gambar 2.1 Elektrolisis asam klorida (HCl) | 6 |
| Gambar 2.2 Sel elektrokimia | 8 |
| Gambar 2.3 Elektroda gelas dan calomel | 10 |
| Gambar 2.4 Elektroda oksigen polarisasi | 11 |
| Gambar 2.5 Skala pH | 13 |
| Gambar 2.6 Hubungan pH - alkalinity | 13 |
| Gambar 2.7 Blok diagram PPI 8255 | 17 |
| Gambar 2.8 Mode Format Kontrol Word | 19 |
| Gambar 2.9 Pembanding Tak Membalik | 20 |
| Gambar 2.10 Pembanding dengan histerisis | 21 |
| Gambar 2.11 Penguat Differensial | 21 |
| Gambar 2.12 Low Pass Filter -40 dB/decade | 22 |
| Gambar 2.13 Penguat Intrumentasi | 23 |
| Gambar 2.14 Blok diagram Successive Approximation ADC | 26 |
| Gambar 3.1 Fungsi Membership Boolean dan Fuzzy | 27 |
| Gambar 3.2 Istilah-istilah dalam Fungsi Membership | 28 |
| Gambar 3.3 Fungsi S | 28 |
| Gambar 3.4 Fungsi π | 29 |
| Gambar 3.5 Fungsi T | 29 |
| Gambar 3.6 Tipikal Sistem Kontrol Dengan Fuzzy Logic | 30 |
| Gambar 3.7 Struktur Dasar Fuzzy Logic Control | 31 |
| Gambar 3.8 Blok diagram NLX230 | 33 |

| | |
|---|----|
| Gambar 3.9 Fungsi Inclusive dan Exclusive | 35 |
| Gambar 3.10 Pin-pin NLX230 | 39 |
| Gambar 4.1 Blok Diagram Sistem | 44 |
| Gambar 4.2 Buffer data NLX230 | 45 |
| Gambar 4.3 Komparator channel | 45 |
| Gambar 4.4 Rangkaian antar-muka paralel | 46 |
| Gambar 4.5 Address Decoder ADC | 47 |
| Gambar 4.6 Rangkaian ADC0804 | 48 |
| Gambar 4.7 Multiplexer Analog | 49 |
| Gambar 4.8 Instrumentasi pH | 50 |
| Gambar 4.9 Instrumentasi DO | 50 |
| Gambar 4.10 Rangkaian low pass filter | 51 |
| Gambar 4.11 Instrumentasi kejernihan air | 52 |
| Gambar 4.12 Filter low pass | 52 |
| Gambar 4.13 Rangkaian Driver Katup | 53 |
| Gambar 4.14 Diagram Alir Program | 54 |
| Gambar 4.15 Diagram Alir Download | 56 |
| Gambar 4.16 Diagram Alir Verify | 57 |
| Gambar 4.17 Diagram Alir Running FMC | 58 |
| Gambar 4.18 Diagram Alir Pembacaan ADC | 59 |
| Gambar 5.1 Grafik Penetralkan dari Keadaan Basa | 66 |
| Gambar 5.2 Grafik Penetralkan dari Keadaan Asam | 67 |

DAFTAR TABEL

| | |
|---|----|
| Tabel 2.1 Potensial oksidasi standar | 9 |
| Tabel 2.2 Tabel kebenaran multiplexer analog 4051 | 24 |
| Tabel 4.1 Channel yang Aktif | 49 |
| Tabel 4.2 Kontrol Download | 55 |
| Tabel 4.3 Kontrol Verify | 55 |
| Tabel 4.4 Matriks Defuzzifier | 59 |
| Tabel 5.1 Hasil Pengukuran pH | 62 |
| Tabel 5.2 Hasil Pengukuran Kejernihan | 63 |
| Tabel 5.3 Hasil Pengukuran DO | 64 |
| Tabel 5.4 Penetralkan Dari Keadaan Basa | 65 |
| Tabel 5.5 Penetralkan Dari Keadaan Asam | 66 |
| Tabel 5.6 Data Volume Penambahan | 68 |

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi dewasa ini sangat pesat dan meliputi segala disiplin ilmu. Teknologi elektronika juga merupakan salah satu disiplin ilmu yang berkembang sangat pesat. Dalam era globalisasi ini, perkembangan teknologi elektronika telah menyentuh semua bidang kehidupan dan berkaitan dengan teknologi-teknologi lainnya. Kehadiran teknologi elektronika telah membuat kegiatan-kegiatan atau proses-proses yang dulunya dilakukan secara manual menjadi suatu proses yang otomatis, cepat, akurat serta lebih efisien.

Kemajuan-kemajuan dalam bidang teknologi elektronika yang menuju ke arah sistem digital serta rangkaian terpadu telah mendorong terciptanya prosesor-prosesor baru yang semakin cepat dengan kemampuan proses yang tinggi dan akurat serta bentuk yang semakin kecil, tetapi dengan kemampuan yang lebih besar. Dengan kemajuan teknologi elektronika ini, maka pekerjaan-pekerjaan dengan konvensional secara bertahap dapat digantikan dengan sistem yang lebih canggih.

Penerapan teknologi elektronika dalam bidang teknologi yang lain, dalam hal ini teknologi kimia sangat banyak, terutama dalam hal kimia instrumentasi. Proses-proses pengukuran dan pengambilan data dari proses-proses kimia yang dulunya dilakukan secara konvensional, saat ini sudah dapat digantikan dengan cara lebih canggih dan otomatis serta memiliki keakurasian yang tinggi sehingga prosesnya lebih cepat dan efisien.

Salah satu penerapan teknologi elektronika dalam bidang kimia instrumentasi adalah dalam hal pengecekan kualitas air baku minum dan air minum yang siap didistribusikan. Proses pengecekan selama ini dilakukan secara konvensional yang membutuhkan waktu dan tenaga

yang cukup banyak, sehingga hal ini menjadi tidak efisien. Dengan pemakaian alat-alat elektronika maka sistem pengecekan kualitas air baku minum dan air minum dapat dipercepat dan lebih efisien.

1.2 Permasalahan

Air minum yang memenuhi persyaratan kesehatan adalah air yang tidak berwarna, tidak berbau, dan tidak berasa. PDAM sebagai penyuplai kebutuhan air minum selalu melakukan pengecekan (*quality control*) kualitas air, baik air baku minum yang berasal dari sungai atau pun air olahan (air minum) yang siap didistribusikan ke pelanggan.

Untuk mengecek kualitas air yang lebih cepat dan lebih efisien maka direncanakan suatu peralatan yang dapat mendeteksi parameter-parameter kualitas air, dalam hal ini parameter pH, tingkat kejernihan, dan kadar oksigen terlarut, dengan menggunakan metode logika Fuzzy. Jika hasil pendeteksian menunjukkan keadaan kualitas air berada di luar daerah optimum yang telah ditentukan, maka secara otomatis peralatan akan melakukan pengkondisian sampai dicapai kondisi yang diinginkan.

1.3 Batasan Masalah

Alat yang direncanakan ini dibatasi untuk mendeteksi kualitas air dalam tiga parameter yaitu kadar oksigen terlarut (*DO/dissolved oxigen*), tingkat asam dan basa (PH), serta tingkat kejernihan. Alat pendeteksi diinterfacekan dengan komputer dan pengambilan keputusan berdasar parameter terukur dengan menggunakan logika fuzzy.

1.4 Tujuan

Perencanaan dan pembuatan alat **APLIKASI FUZZY LOGIC CONTROL PADA ALAT**

QUALITY CONTROL AIR MINUM PDAM YANG DIINTERFACEKAN KE KOMPUTER, bertujuan agar sistem kontrol kualitas air PDAM menjadi lebih cepat dan lebih efisien.

Selain tujuan di atas, perencanaan dan pembuatan peralatan ini bertujuan untuk mempelajari proses akuisisi data, proses pengambilan keputusan dengan menggunakan logika fuzzy.

1.5 Metodologi

Pengerjaan Tugas Akhir ini dilakukan dengan langkah-langkah sebagai berikut :

- ◆ Studi literatur mengenai kontrol kualitas air baku minum dan air minum yang meliputi: pengukuran DO, PH, dan tingkat kejernihan, chip fuzzy generasi NLX230, *Analog to Digital Converter* ADC0804, IBM-PC, Elektronika industri, pemrograman komputer serta rangkaian elektronika.
- ◆ Perencanaan perangkat keras dan perangkat lunak dari peralatan.
- ◆ Pembuatan perangkat keras dan perangkat lunak sesuai dengan perencanaan yang telah dibuat.
- ◆ Pengujian dan pengkalibrasian peralatan yang dibuat serta pengukuran data dari hasil pemantauan peralatan.
- ◆ Penulisan buku Tugas Akhir.

1.6 Sistematika

Dalam buku tugas akhir ini, pembahasan mengenai peralatan yang dibuat dibagi dalam enam bab dengan sistematika sebagai berikut:

Bab I berisi pendahuluan yang membahas latar belakang, tujuan, permasalahan, dan pembatasan masalah, metodologi serta sistematika penulisan buku tugas akhir.

- Bab II berisi penjelasan mengenai teori dasar dari proses kimia reaksi pH, DO, dan kejernihan, transduser, *analog to digital converter*, operasional amplifier.
- Bab III berisi teori dasar logika fuzzy dan chip fuzzy tipe NLX230.
- Bab IV berisi perencanaan perangkat keras dan perangkat lunak yang menyangkut blok diagram, diagram alur, serta prosedur-prosedur yang digunakan dalam perangkat lunak.
- Bab V berisi pengukuran dan kalibrasi dari sensor yang digunakan dan keterbatasan pada sensor yang digunakan.
- Bab VI berisi penutup yang terdiri dari kesimpulan yang diperoleh dari pembuatan Tugas Akhir ini, serta saran-saran untuk pengembangan dan penerapan Tugas Akhir ini.

Penyelesaian dari tugas akhir ini dilakukan dengan metode/langkah-langkah sebagai berikut:

- ◆ Penelitian di lapangan untuk mencari data-data tentang quality control air minum dan cara kerja quality control air minum PDAM.
- ◆ Studi literatur mengenai sistem akuisisi data, penggunaan metode algoritma Fuzzy pada sistem kontrol, pengukuran derajat keasaman (pH), kadar oksigen terlarut (DO), tingkat kejernihan, analog to digital converter ADC0804, pemrograman komputer dan rangkaian elektronika.
- ◆ Perencanaan perangkat keras dan lunak peralatan.
- ◆ Pembuatan perangkat keras dan perangkat lunak sesuai dengan perencanaan yang telah dibuat.
- ◆ Pengujian dan pengkalibrasian peralatan yang telah dibuat serta melakukan pengukuran data dari parameter-parameter pH, DO, dan tingkat kejernihan.

- ◆ Penulisan buku Tugas Akhir.

1.7 Relevansi

Dengan selesainya alat ini diharapkan dapat dikembangkan dan disempurnakan sesuai dengan kebutuhan dan dapat diterapkan pada instansi yang mempunyai permasalahan yang sama.

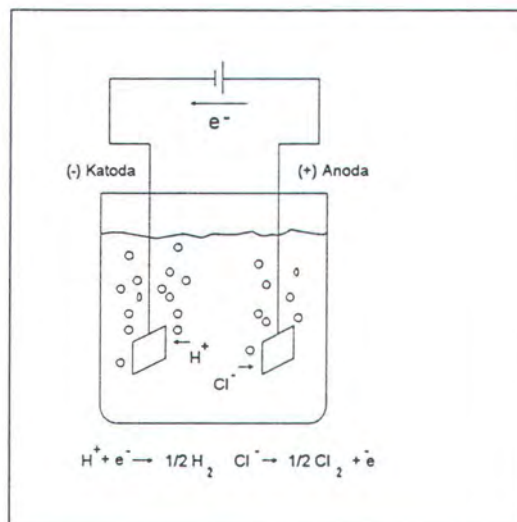
BAB II

TEORI PENUNJANG

2.1 Elektrokimia

2.1.1 Arus Dan Perubahan Kimia

Jika suatu elektroda dicelupkan ke dalam suatu larutan, maka akan timbul arus yang mengalir melalui larutan dari elektroda satu menuju ke elektroda lain. Hal ini terjadi disebabkan adanya perubahan kimia pada elektroda. Sifat dari perubahan kimia tergantung pada komposisi larutan, sifat elektroda, dan besarnya gaya gerak listrik yang diberikan.



Gambar 2.1¹

Elektrolisis asam klorida (HCl)

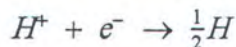
Sebagai contoh, proses perubahan kimia yang terjadi pada elektroda platinum yang dicelupkan ke dalam larutan HCl, seperti terlihat pada gambar 2.1. Jika diberikan tegangan, misal 1,3 Volt, maka ditemukan H₂ akan menempel pada katoda dan Cl₂ pada anoda.

¹ Sawyer, Clair N., CHEMISTRY FOR SANITARY ENGINEERS, second edition, McGraw-Hill Book Company, Kogakusha Company Ltd., 1987, hal. 186

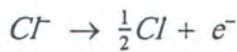
Gerakan arus yang terjadi adalah sebagai berikut :

Gaya gerak listrik yang diterapkan mengakibatkan adanya pergerakan elektron. Elektron akan mengalir melalui bagian luar konduktor logam dengan arah seperti pada gambar di atas. Aliran tersebut akan mempertahankan muatan negatif pada katoda dan muatan positif pada anoda. Sedangkan aliran arus melalui larutan terjadi karena pergerakan dari kation (H^+) ke muatan negatif katoda dan dari anion (Cl^-) ke muatan positif anoda.

Pada saat ion H^+ mencapai katoda, maka ion H^+ akan mengikat elektron dan direduksi menjadi gas H_2 sesuai dengan reaksi :

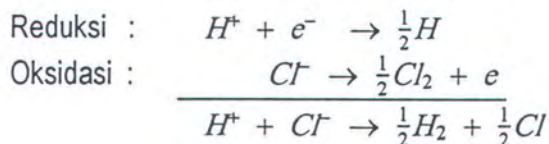


Pada saat ion Cl^- mencapai anoda, maka ion Cl^- akan memberikan elektron dan dioksidasi menjadi gas Cl_2 sesuai dengan reaksi berikut :



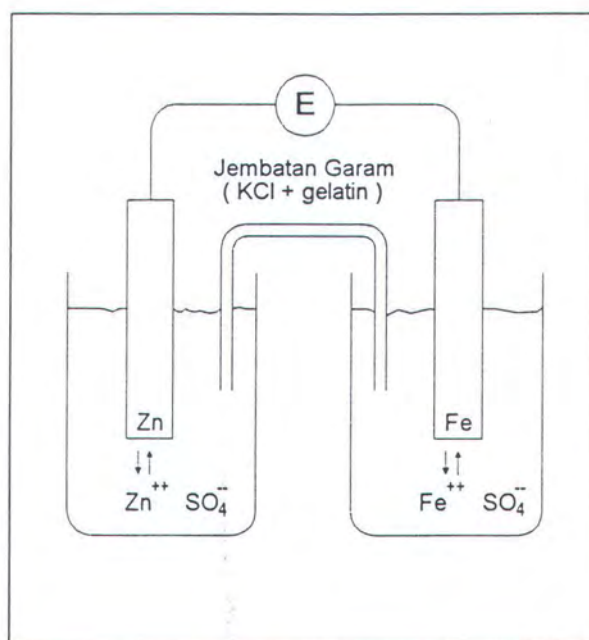
Elektron yang dilepaskan ion Cl^- akan dialirkan melalui rangkaian luar oleh gaya gerak listrik baterai, sehingga dapat diikat ion H^+ pada katoda. Seperti terlihat pada persamaan reaksi di atas, peristiwa reduksi akan terjadi pada katoda dan oksidasi terjadi pada anoda.

Perubahan kimia secara keseluruhan yang terjadi pada larutan adalah sebagai berikut :



2.1.2 Sel Elektrokimia

Sebuah elektroda yang dicelupkan pada larutan disebut dengan istilah setengah sel. Sedangkan kombinasi dua buah setengah sel yang dihubungkan, sehingga ion dapat melewatinya disebut sel elektrokimia atau sel galvanik.



Gambar 2.2²
Sel elektrokimia

Sel elektrokimia hanya memerlukan satu jenis larutan, tetapi larutan untuk menggabungkan kedua setengah sel berbeda dan harus dicegah jangan sampai bercampur. Biasanya digunakan beberapa tipe jembatan garam, sehingga dimungkinkan untuk melewatkan ion dan tetap menjaga interdifusi larutan secara minimum, seperti terlihat pada gambar 2.2.

Jika sebuah voltmeter dipasang pada ujung-ujung setengah sel, seperti pada gambar 2.2, maka sel akan menghasilkan gaya gerak listrik (emf). Gaya gerak listrik (emf) ini dapat dihitung dari tabel 2.1 yang berdasarkan pada elektroda hidrogen standar, yang sesuai dengan perjanjian bernilai $E = E^0 = 0$. Dengan menghitung perbedaan potensial standar dari dua setengah sel, maka potensial sel secara keseluruhan dapat ditentukan.

² Ibid, hal. 191

Tabel 2.1³
Potensial oksidasi standar

| Reaksi setengah sel | E° (Volt) |
|--|-----------|
| $Mg + 2OH^- \rightarrow Mg(OH)_2 + 2e^-$ | 2,690 |
| $Mg \rightarrow Mg^{++} + 2e^-$ | 2,370 |
| $Al \rightarrow Al^{++} + 3e^-$ | 1,660 |
| $Zn + S^{\pm} \rightarrow ZnS + 2e^-$ | 1,440 |
| $Zn + 2OH^- \rightarrow Zn(OH)_2 + 2e^-$ | 1,245 |
| $Zn \rightarrow Zn^{++} + 2e^-$ | 0,763 |
| $S^{\pm} \rightarrow S + 2e^-$ | 0,480 |
| $Fe \rightarrow Fe^{++} + 2e^-$ | 0,441 |
| $Sn \rightarrow Sn^{++} + 2e^-$ | 0,136 |
| $Pb \rightarrow Pb^{++} + 2e^-$ | 0,126 |
| $H_2 \rightarrow 2H^+ + 2e^-$ | 0,000 |
| $H_2S \rightarrow S + 2H^+ + 2e^-$ | -0,141 |
| $Ag + Cl^- \rightarrow AgCl + e^-$ | -0,222 |
| $Cu \rightarrow Cu^{++} + 2e^-$ | -0,337 |
| $2Ag + CrO_4^- \rightarrow Ag_2CrO_4 + 2e^-$ | -0,446 |
| $Fe^{++} \rightarrow Fe^{3+} + e^-$ | -0,771 |
| $Ag \rightarrow Ag^+ + e^-$ | -0,799 |
| $2H_2O \rightarrow O_2 + 4H^+ + 4e^-$ | -1,229 |

2.1.3 Elektroda

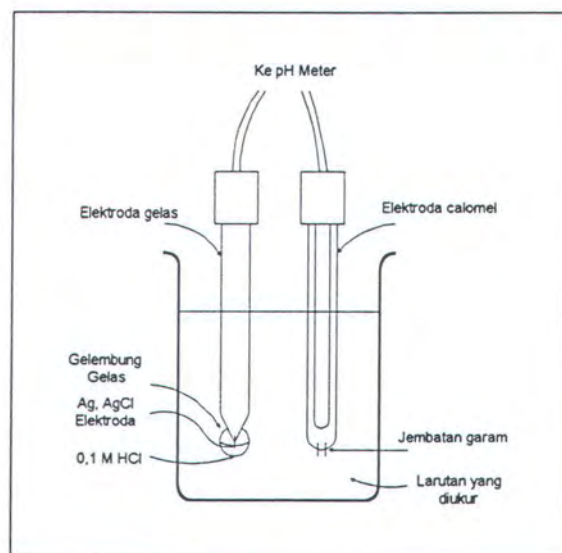
Secara garis besar elektroda diklasifikasikan menjadi dua kelas, yaitu elektroda indikator dan referensi. Elektroda referensi mempunyai persyaratan harus mudah dibuat, tetap konstan, dan menghasilkan tegangan walaupun hanya melewatkan arus yang kecil. Sedangkan elektroda indikator harus mempunyai respon terhadap perubahan aktivitas dari spesies yang diukur.

Dalam tugas akhir ini akan dibahas elektroda-elektroda yang digunakan saja, yaitu elektroda *calomel*, elektroda gelas dan elektroda oksigen polarisasi.

³ Ibid, hal. 192

2.1.3.1 Elektroda Calomel (SCE)

Elektroda ini merupakan jenis elektroda referensi yang terbuat dari larutan yang telah dijenuhkan dengan *calomel* (merkuri klorida) dan potasium klorida dan ditempatkan di atas lapisan merkuri, seperti pada gambar 2.3.



Gambar 2.3⁴
Elektroda gelas dan calomel

2.1.3.2 Elektroda Gelas

Elektroda gelas terdiri dari gelembung gelas yang sensitif pH pada ujungnya, yang berisi larutan klorida yang diketahui pHnya dan elektrode referensi, biasanya perak/perak klorida atau *calomel*. Bahan gelas terbuat dari komposisi gelas khusus, yaitu sebagian dihidrasi aluminium silikon yang mengandung ion sodium atau kalsium dan sebagian kecil ion lantanida.

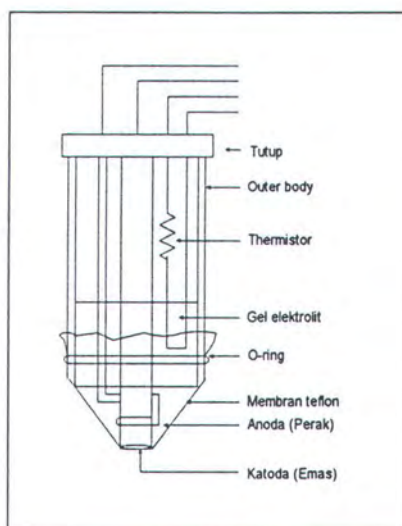
Bagian dalam gelas akan kontak dengan larutan 0,1 M HCl dan bagian luar kontak dengan larutan yang diukur pHnya. Pada setiap permukaan, membran gelas akan menyerap air dan membentuk lapisan gel. Ion hidrogen larutan dapat berdifusi melalui lapisan gel dan

⁴ Pescok, Robert L., MODERN METHODS OF CHEMICAL ANALYSIS, second edition, John Wiley & Sons Inc., 1986, hal. 379

mengganti (pertukaran ion) ion sodium atau logam lain dalam struktur gelas. Hasil akhir dari proses difusi dan pertukaran ion adalah fase pengikatan tegangan pada kedua sisi membran gelas akan berubah dan besarnya ditentukan oleh aktivitas ion hidrogen dalam larutan yang diukur.

2.1.3.3 Elektroda Membran

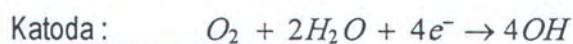
Molekul oksigen dalam larutan dapat diukur dengan elektroda membran yang sering disebut dengan sensor *Clark* oksigen. Anoda yang terbuat dari perak dan katoda yang terbuat dari emas dihubungkan secara listrik oleh potasium klorida atau larutan elektrolit lain. Elektroda secara keseluruhan dipisahkan dengan sampel oleh membran tipis teflon (PTFE) atau polyethylene yang permeable terhadap oksigen.



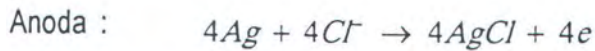
Gambar 2.4⁵

Elektroda oksigen polarisasi

Jika tegangan sebesar 800 mV diberikan di antara anoda dan katoda, maka reaksi yang terjadi jika terdapat oksigen adalah :



⁵ Ibid, hal. 538



Besarnya arus yang dihasilkan adalah sebanding dengan jumlah oksigen yang terkandung di dalam sampel. Oleh karena kecepatan difusi oksigen melalui membran bervariasi sekitar 3% per $^{\circ}C$, maka elektroda ini dilengkapi dengan thermistor yang secara otomatis akan menyetel sinyal output untuk dikompensasi terhadap perubahan temperatur. Kecepatan respon dari sensor ditentukan oleh kecepatan difusi oksigen, biasanya 90% dari respon akhir dan terjadi selama 10 detik.

2.1.4 Pengukuran pH

Pengukuran pH digunakan untuk menentukan tingkat keasaman atau kebasaan atau lebih tepatnya untuk mengetahui konsentrasi ion hidrogen bebas dalam suatu larutan.

Hasil pengukuran konsentrasi ion hidrogen dengan elektroda hidrogen terhadap air murni adalah sama dengan 10^{-7} mole/l pada saat kesetimbangan.



Oleh karena air hanya menghasilkan satu ion hidroksil untuk tiap ion hidrogen, maka 10^{-7} mole hidroksil juga dihasilkan secara bersamaan. Dengan substitusi ke persamaan aksi massa kita peroleh :

$$\frac{[H^+][OH^-]}{[H_2O]} = K$$

Tetapi karena konsentrasi air sangat besar sehingga persamaan di atas dapat diubah menjadi :

$$[H^+][OH^-] = K$$

di mana K_w = konstanta ionisasi air

Konstanta ionisasi dari air adalah :

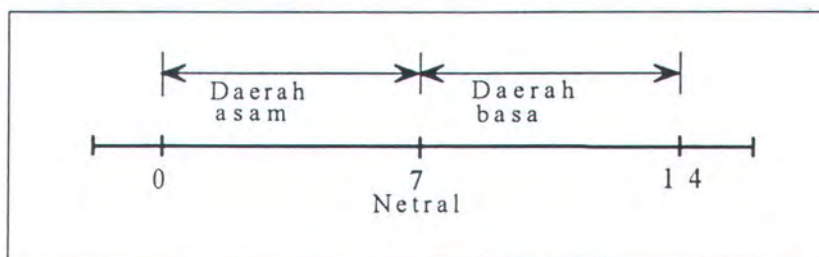
$$[H^+][OH^-] = 10^{-7} \times 10^{-7} = 10^{-14}$$

Untuk memudahkan perhitungan, maka diambil logaritma negatifnya yaitu :

$$pH = -\log [H^+] \text{ atau } pH = \log \frac{1}{[H^+]}$$

$$pH = \log \frac{1}{[H^+]} = \log \frac{1}{[10^{-7}]} = 7$$

Skala pH adalah berkisar dari 0 sampai 14, dimana pada saat $\text{pH}=7$ secara mutlak menunjukkan daerah netral. Kondisi asam akan bertambah jika nilai pH berkurang, sedangkan kondisi basa bertambah jika nilai pH bertambah, seperti terlihat pada gambar 2.5.

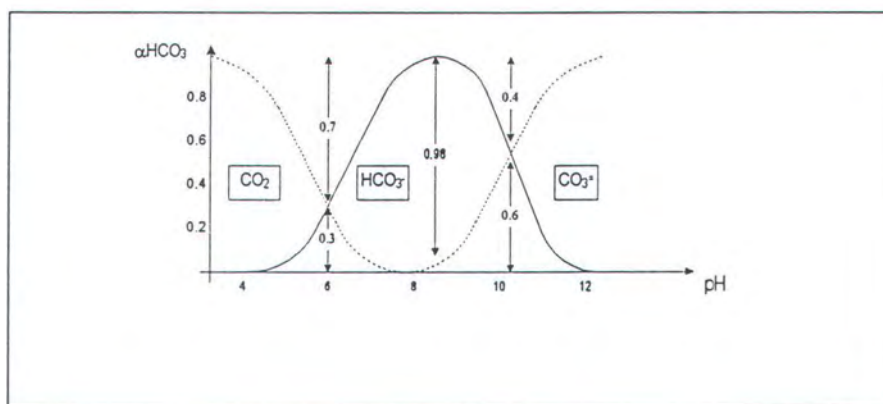


Gambar 2.5

Skala pH

2.1.4.1 Hubungan pH Dan Alkalinitas

Air mempunyai sifat alkalinitas yang disebabkan adanya unsur-unsur karbonat (CO_3), bicarbonat (HCO_3), hidroksida, dan ion OH^- . Tetapi juga sering terdapat unsur fosfat, ion silikat, dan beberapa unsur lain yang turut mempengaruhi alkalinitas air. Gambar 2.6 memperlihatkan air yang mengandung unsur CO_3 , HCO_3 , OH^- .

Gambar 2.6⁶

Hubungan pH - alkalinity

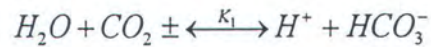
⁶

Fair, Gordon M., Geyer, John C., Okun, Daniel A, WATER AND WASTEWATER ENGINEERING, John Wiley and Sons, Inc., hal. 28-22

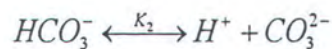
Dari grafik di atas terjadi peristiwa:

- ♦ CO_2 ; dominan pada pH rendah, hilang pada pH tinggi.
- ♦ HCO_3^- ; mulai ada pada pH 7,5, hilang pada pH tinggi.
- ♦ CO_3^{2-} ; mulai ada pada pH 9.

Reaksi kesetimbangan dalam air sebagai berikut:



$$\text{dimana } K_1 = \frac{(\text{H}^+)(\text{HCO}_3^-)}{(\text{CO}_2)}$$



$$\text{dimana } K_2 = \frac{(\text{H}^+)(\text{CO}_3^{2-})}{\text{HCO}_3^-}$$

2.1.5 Pengukuran DO

DO (*dissolved oxygen*) didefinisikan sebagai banyaknya oksigen terlarut per satuan liter cairan. Pengukuran DO dilakukan dengan cara mencelupkan transduser DO tipe HI 76401 ke dalam air yang diukur. Transduser ini mengeluarkan output arus sebesar 0,1 nA pada saat *zero oxygen* dan 100 nA pada saat oksigen jenuh.

2.1.5.1 Pengukuran Dissolved Oxygen (DO) Dengan Metode Winkler

Metode menentukan oksigen terlarut dalam air ada beberapa cara. Salah satu metode itu adalah metode Winkler. Adapun metode Winkler adalah sebagai berikut:

Alat-alat percobaan:

- | | |
|--------------------------------------|---------|
| ♦ Botol Winkler dengan volume 300 cc | 1 buah. |
| ♦ Pipet 10 cc dilengkapi pompa | 3 buah. |
| ♦ Buret | 1 buah. |

- ♦ Slang plastik 1 buah.
- ♦ Jerigen penampung air contoh 2 buah.

Bahan-bahan:

- ♦ Reagen-reagen:
 - ✓ NaOH/KY.
 - ✓ MnCl_2 .
 - ✓ H_3PO_4 .
- ♦ Tio Sulfat dengan faktor 9,71.
- ♦ Amilum.
- ♦ Air olahan IPA Ngagel (air contoh).
- ♦ Air baku/sungai.

Langkah-langkah percobaan penentuan DO air contoh dan air baku:

1. Air diambil dari jirigen dengan menggunakan slang plastik dimasukkan ke dalam botol Winkler. Posisi ujung-ujung slang harus berada di dasar jerigen dan botol Winkler. Selama pengambilan ini aliran air tidak boleh terputus oleh udara. Jika terjadi harus diulang dari awal. Botol diisi sampai penuh dan botol ditutup.
2. Tutup dibuka, kemudian ditambahkan reagent-reagent:
 - ♦ NaOH sebanyak 2 cc.
 - ♦ MnCl_2 sebanyak 2 cc.

Botol ditutup kembali dan dikocok. Larutan menjadi keruh (berwarna). Botol diletakkan dan ditunggu sampai terjadi pemisahan/terjadi pengendapan.
3. Setelah terjadi pengendapan, air bening sebelah atas dibuang perlahan dan endapan disisakan. Endapan ditambah dengan reagent H_3PO_4 sebanyak 4 cc dan dikocok sampai

didapat larutan bening berwarna. Larutan disimpan ke dalam ruang gelap selama 10 menit.

4. Langkah berikutnya larutan dititrasi dengan Thio Sulfat yang sudah diketahui faktornya. Titrasi dihentikan setelah didapat larutan yang bening. Larutan bening ini dites dengan amilum (sebagai indikator). Larutan berubah warna menjadi kebiruan. Kemudian dititrasi lagi dengan Thio Sulfat sampai bening kembali dan dites lagi dengan Amilum. Jika larutan tetap bening, maka proses titrasi dihentikan dan dibaca volume Thio Sulfat yang digunakan.
5. DO (disolved oxigen) air contoh dihitung dengan rumus:

$$DO = \frac{\text{vol. titer} \times \text{faktor Thio Sulfat} \times 80}{\text{vol. botol Winkler} - 4 \text{ cc}}$$

2.1.6 Pengukuran Tingkat Kejernihan

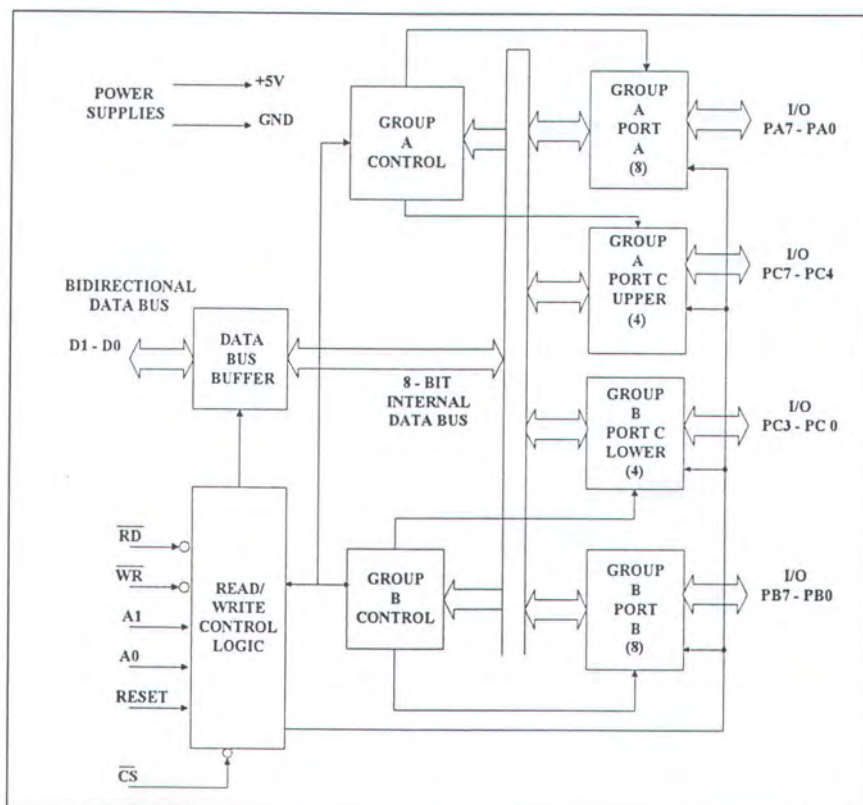
Pengukuran tingkat kejernihan dilakukan dengan mencelupkan alat yang terdiri dari fotodiode dan lampu sebagai sumber cahaya serta air sebagai media rambat cahaya. Dengan mengetahui karakteristik hubungan resistansi fotodiode terhadap intensitas cahaya yang mengenainya, maka fotodiode dan lampu dikonfigurasi sehingga dapat berfungsi untuk mendeteksi tingkat kejernihan air.

Intensitas cahaya yang diterima fotodiode dipengaruhi kejernihan air. Semakin jernih air, maka intensitas cahaya yang diterima fotodiode cukup tinggi yang selanjutnya resistansi fotodiode mengecil. Begitu pula sebaliknya. Dengan demikian tingkat kejernihan air berpengaruh langsung terhadap resistansi fotodiode.

2.2 Programmable Peripheral Interface (PPI) 8255

PPI 8255 merupakan paralel I/O yang dapat diprogram untuk mentransfer data dalam berbagai kondisi, dari I/O yang sederhana sampai dengan *interrupt* I/O yang kompleks. PPI 8255 mempunyai tiga buah port paralel 8 bit, yaitu *port A*, *port B*, dan *port C*. Untuk *port C* dapat dipisah menjadi 4 bit orde rendah dan 4 bit orde tinggi. Fungsi masing-masing port dapat diprogram sebagai input/output dengan inisialisasi *Control Word* melalui *Control Register*.

PPI 8255 dapat diklasifikasikan dalam 2 mode, yaitu : *mode Bit Set/Reset (BSR)* dan *mode I/O*. *Mode BSR* digunakan untuk menset/reset bit dalam port, sedangkan *mode I/O* dibagi menjadi tiga, yaitu : *mode 0*, *mode 1*, dan *mode 2*. Block diagram PPI 8255 diperlihatkan dalam gambar 2.7.



Gambar 2.7⁷

Blok diagram PPI 8255

⁷ Hall, Douglas V., MICROPROCESSOR AND INTERFACING, McGraw-Hill Book Co., 1986, hal. 263

Pada bagian control terdapat 6 jalur pengontrol yaitu :

- RD (Read)

Sinyal kontrol yang memungkinkan operasi pembacaan. Jika *low*, PPI 8255 di-*enable* dan akan mengirimkan data atau status ke data bus.

- WR (Write)

Sinyal kontrol yang memungkinkan operasi penulisan data atau kontrol word, dimana sinyalnya aktif *low*.

- RESET

Sinyalnya aktif *high*, dimana sinyal ini akan mereset kontrol register dan semua port di-set ke dalam mode input.

- CS (Chip Select)

Merupakan sinyal pemilih yang digunakan untuk mengaktifkan PPI 8255 dimana sinyalnya aktif *low*.

- ADDRESS INPUT (A0 - A1)

Kombinasi dari kedua address input ini menentukan register dari PPI 8255 yang akan menerima atau mengirim data dari atau ke mikroprosesor.

2.2.1 Pengoperasian PPI 8255

Mode kerja dari PPI 8255 dioperasikan dalam tiga mode, yaitu :

- Mode 0 : *Basic Input / Output*

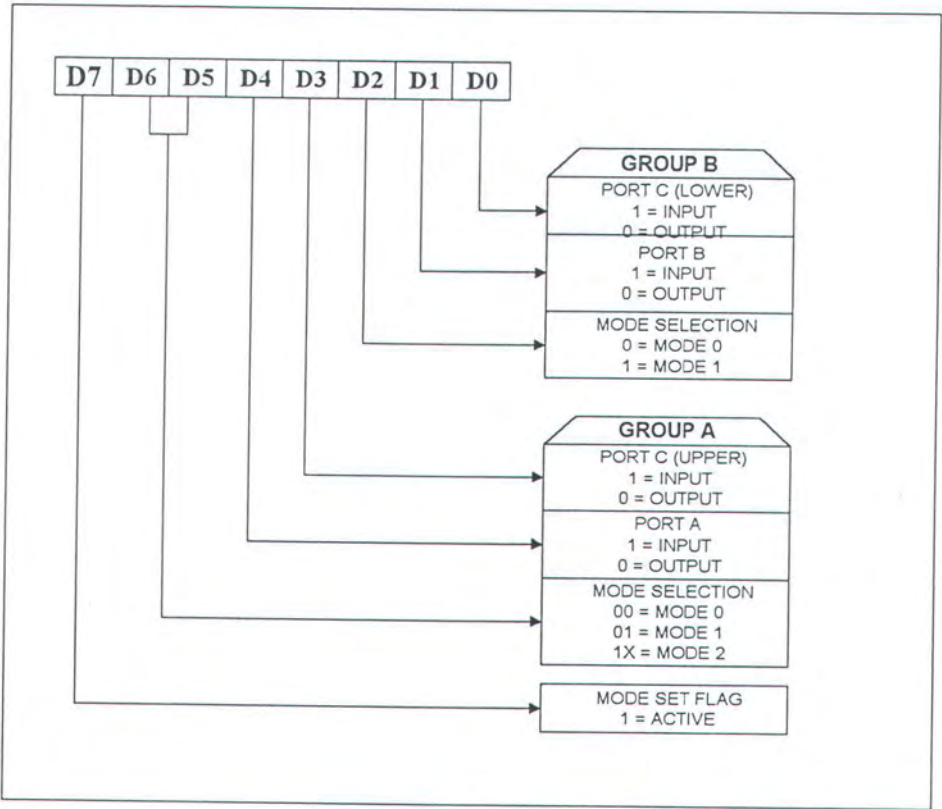
- Mode 1 : *Strobed Input / Output*

- Mode 2 : *Bi-Directional Bus*

Pada gambar 2.8 diperlihatkan definisi bit dari control register, dimana isi tiap bit dari control register 8 bit ini akan menentukan fungsi dari masing-masing port.

Dari gambar terlihat bahwa bit D0 sampai D2 merupakan bagian grup B. Bit D0 mengatur 4 bit bawah dari port C, yaitu sebagai input jika logikanya *high* dan output jika *low*.

Bit D1 mengatur fungsi port B sebagai input jika logikanya *high* dan output jika *low*. Bit D2 menentukan mode operasi dari port B dimana terdapat dua mode operasi, yaitu : *mode 0* jika logikanya *low* dan *mode 1* jika *high*.



Gambar 2.8⁸
Mode format kontrol word

Bit D3 sampai D6 merupakan bagian grup A dimana bit D3 menentukan operasi kerja 4 bit atas port C dan bit D4 untuk operasi kerja port A. Untuk bit D5 dan D6 berfungsi untuk memilih salah satu dari 3 mode operasi, yaitu : *mode 0*, *mode 1*, dan *mode 2*.

Bit D7 merupakan *mode set flag* yang harus *high* jika mode operasi akan diubah.

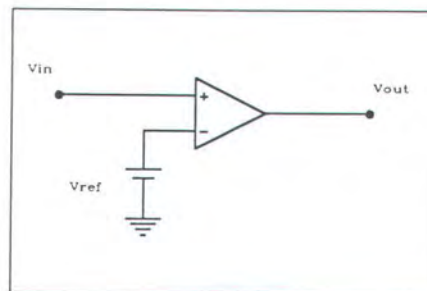
⁸ Ibid, hal. 265

2.3 Penguat Operasional

Keluaran transduser sangat kecil, sehingga perlu dikuatkan sampai mencapai jangkauan tegangan input ADC skala penuh dengan menggunakan rangkaian pengkondisi sinyal. Penguat operasional merupakan komponen utama dari rangkaian pengkondisi sinyal. Penerapan pengkondisi sinyal dalam rangkaian ini adalah sebagai penguat pembalik, penguat tak membalik, pengikut tegangan, penguat differensial, filter aktif dan penguat instrumentasi.

2.3.1 Pembandingan (Komparator)

Sebuah pembandingan bekerja dengan cara membandingkan sinyal input dengan tegangan referensi. Sinyal input diterapkan pada salah satu masukan dari op-amp sedangkan tegangan referensi diterapkan pada masukan lain.

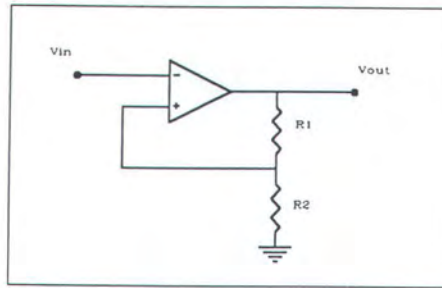


Gambar 2.9
Pembandingan Tak Membalik

Jika sinyal masukan (V_{in}) lebih besar dari tegangan referensi, maka tegangan keluaran (V_{out}) menjadi positif saturasi ($+V_{sat}$), sedangkan jika sinyal masukan lebih kecil dari tegangan referensi, tegangan keluaran menjadi negatif saturasi ($-V_{sat}$).

Pembandingan di atas mempunyai kelemahan jika sinyal masukan sedikit lebih besar atau lebih kecil dari tegangan referensi yang berakibat tegangan output berosilasi antara $+V_{sat}$ dan $-V_{sat}$. Hal ini menjadi riskan untuk sinyal masukan yang mengandung noise. Untuk mengatasi hal

ini, rangkaian di atas ditambah histerisis yaitu menambah umpan balik positif seperti pada gambar di bawah.



Gambar 2.10
Pembanding dengan histerisis

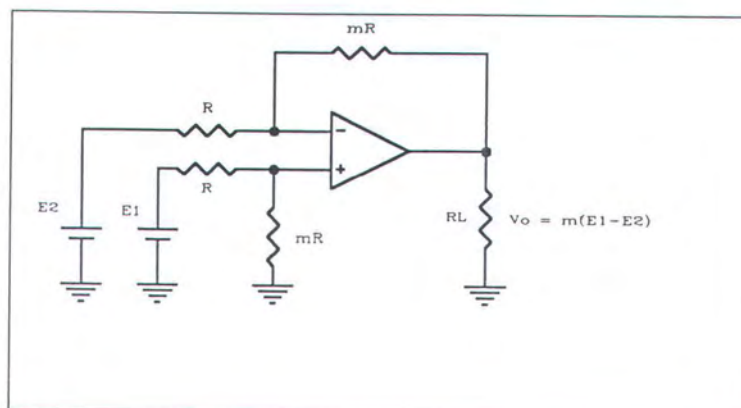
Tegangan histerisis merupakan selisih antara V_{UT} dan V_{LT} dimana :

$$V_{UT} = \frac{R_2}{R_1 + R_2} (+V_{SAT})$$

$$V_{LT} = \frac{R_2}{R_1 + R_2} (-V_{SAT})$$

2.3.2 Penguat Differensial

Penguat differensial berfungsi untuk menguatkan sinyal-sinyal kecil yang mungkin terbenam dalam sinyal yang lebih besar. Rangkaian ini dapat pula difungsikan sebagai rangkaian penjumlah tegangan dengan penguatan tertentu tergantung pada resistor-resistornya.



Gambar 2.11
Penguat Differensial

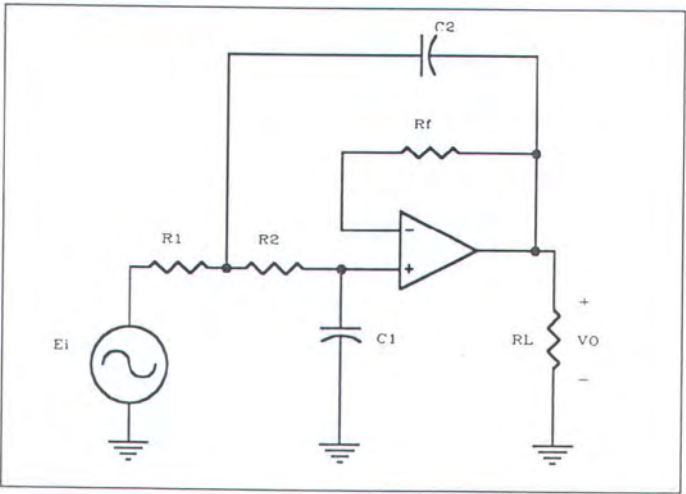
Bila $E_1 = 0$, maka rangkaian berfungsi sebagai penguat pembalik, dimana E_2 akan dikuatkan sebesar $-m$ sehingga output V_o menjadi $-mE_2$. Sedangkan bila $E_2 = 0$, maka rangkaian berfungsi sebagai rangkaian tak membalik, dimana E_1 akan terbagi antara R dan mR sehingga input terminal (+) menjadi $mE_1/(1+m)$ dan dikuatkan sebesar $(m+1)$ sehingga output menjadi mE_1 . Bila kedua sinyal input E_1 dan E_2 merupakan masukan pada terminal (+) dan (-) dari op-amp, maka besar tegangan output adalah :

$$V_o = mE_1 - mE_2 = m(E_1 - E_2)$$

m disebut dengan gain differensial yang merupakan perbandingan resistornya. Persamaan di atas menunjukkan bahwa tegangan output sebanding dengan perbandingan tegangan pada kedua terminal masukannya. Dan jika E_2 dibuat negatif maka rangkaian di atas sebagai penguat penjumlah.

2.3.3 Low Pass Filter

Low pass filter berfungsi untuk meredam sinyal masukan yang mempunyai frekuensi di atas frekuensi cut off. Gambar di bawah menunjukkan rangkaian low pass filter -40 dB/decade jenis *Butterworth*.



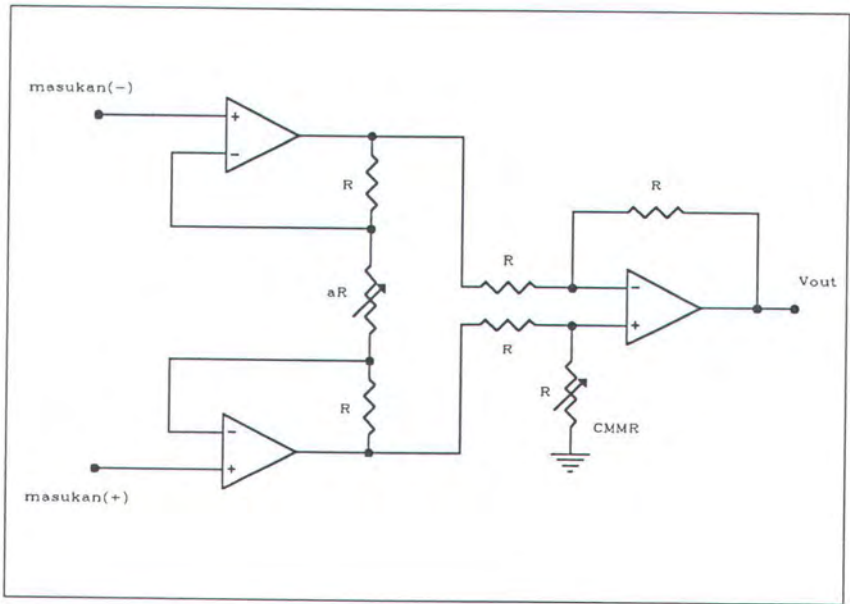
Gambar 4.12
Low Pass Filter -40 dB/decade

Filter ini menghasilkan slope/landaian sebesar -40 dB/dekade yang berarti setiap kenaikan frekuensi sebesar 10 kali menyebabkan penurunan penguatan sebesar 40 dB.

2.3.4 Penguat Instrumentasi

Penguat instrumentasi merupakan penguat differensial tegangan presisi yang diperlukan jika interferensi di sekitarnya cukup tinggi dan dikhususkan terutama untuk pengukuran sinyal-sinyal yang kecil. Penguat instrumentasi dapat dibangun dengan memakai tiga buah op-amp, yaitu dua buah sebagai penyangga (*voltage follower*) dan sebuah sebagai penguat differensial, seperti terlihat pada gambar 2.13. Penguatan tegangan ditentukan oleh besar resistor aR , yaitu:

$$A_V = \frac{V_O}{V_i} = 1 + \frac{2}{a} \quad \text{dimana } a = aR/a$$



Gambar 2.13⁹
Penguat instrumentasi

⁹ Coughlin, Robert F., PENGUAT OPERATIONAL DAN RANGKAIAN TERPADU LINEAR, edisi kedua, Erlangga, 1985, hal. 168

2.4 Pengubahan Sinyal Analog ke Digital

Data yang akan diolah komputer harus merupakan data digital (0 dan 1). Sedangkan sinyal yang berasal dari transduser merupakan sinyal analog, untuk itu diperlukan suatu sistem pengubah sinyal analog menjadi sinyal digital. Sistem pengubah ini biasanya terdiri dari pemilih kanal input (*multiplexer analog*) dan *analog to digital converter* (ADC).

Dalam bagian ini akan dijelaskan mengenai bagian-bagian dari akuisisi data yang terdiri dari penguat instrumentasi, multiplexer, dan *Analog to Digital Converter*.

2.4.1 Multiplexer Analog

Multiplexer analog merupakan rangkaian multipleks yang digunakan untuk memilih sinyal input analog, di mana kanal terpilih dipilih secara digital melalui kontrol kanal. Tabel kebenaran untuk mutiplexser analog 4051 dapat dilihat pada tabel 2.2.

Tabel 2.2¹⁰
Tabel kebenaran multiplexer analog 4051

| Input | | | | " ON " Channel |
|-------|---|---|---|-------------------|
| Inh | C | B | A | |
| 1 | X | X | X | None |
| 0 | 0 | 0 | 0 | Y0 |
| 0 | 0 | 0 | 1 | Y1 |
| 0 | 0 | 1 | 0 | Y2 |
| 0 | 0 | 1 | 1 | Y3 |
| 0 | 1 | 0 | 0 | Y4 |
| 0 | 1 | 0 | 1 | Y5 |
| 0 | 1 | 1 | 0 | Y6 |
| 0 | 1 | 1 | 1 | Y7 |

¹⁰ ----, CMOS DATABOOK, National Semiconductor Corporation, 1984, hal. 1-377

Jangkauan tegangan input analog yang dapat diberikan dapat berupa tegangan input puncak ke puncak (V_{p-p}) atau tegangan DC, tergantung dari besarnya tegangan yang diberikan pada pin V_{DD} dan $-V_{EE}$.

2.4.2 Konversi Analog Ke Digital

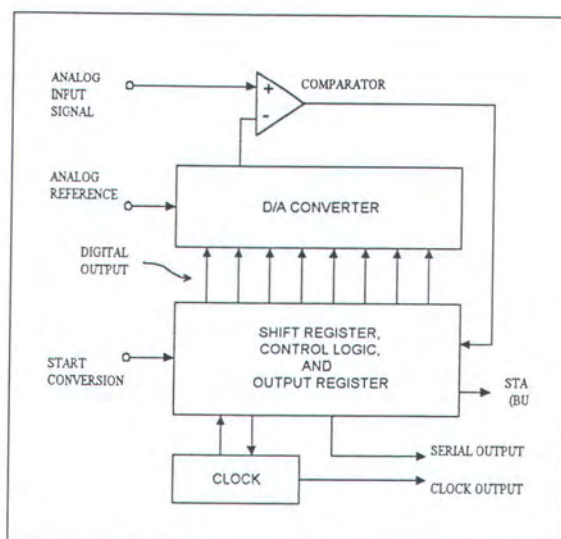
ADC merupakan peralatan yang dipakai untuk mengubah sinyal listrik analog menjadi sinyal digital. Dalam pembuatan ADC digunakan metoda tertentu, yang mana spesifikasinya ditentukan oleh kecepatan konversi (*conversion time*), keakuratan (*accuracy*), stabilitas (*stability*), maupun faktor biaya.

Ada dua macam metoda dalam pembuatan ADC, yaitu dengan metoda loop terbuka dan metoda loop tertutup. Dengan metoda loop terbuka dikenal *Flash ADC*, *Time Window ADC*, *Single Ramp Integrating* dan *Dual Slope Integrating*. Sedangkan dengan metode loop tertutup dikenal tipe-tipe, seperti *Single Tracking ADC* dan *Successive Approximation ADC*. Dalam Tugas Akhir ini akan digunakan ADC jenis *Successive Approximation ADC*.

2.4.2.1 Successive Approximation ADC

ADC jenis ini mempunyai rangkaian yang lebih kompleks dibandingkan dengan jenis *Ramp Integrating*, tetapi mempunyai waktu konversi yang lebih kecil. ADC jenis ini tidak menggunakan pencacah untuk memberikan masukan ke blok pengubah digital ke analognya (*DAC*), tetapi menggunakan *Register Successive Approximation (SAR)*, seperti terlihat pada gambar 2.14.

Cara kerjanya adalah sebagai berikut :

Gambar 2.14¹¹

Blok diagram Successive Approximation ADC

- MSB diset *high* dan bit-bit lainnya diberi logika *low*. Hal ini akan menghasilkan nilai V_{out} lebih besar daripada V_{in} , sehingga keluaran pembanding berubah menjadi *low* dan akan mereset MSB. Sebaliknya jika V_{out} lebih kecil dari V_{in} , MSB tetap *high*.
- MSB kedua diset *high*, dan jika nilai baru dari V_{out} lebih besar dari V_{in} , maka bit ini akan direset menjadi *low* dan sebaliknya jika lebih kecil, bitnya akan tetap *high*.
- Proses tersebut akan terus dilakukan untuk seluruh bit dalam register, dimana diperlukan satu periode *clock* untuk setiap bit. Hal ini merupakan kelebihan dari *SA-ADC* sehingga waktu konversinya cepat. Dengan resolusi 12 bit akan dibutuhkan 12 pulsa *clock* untuk satu kali konversi.

¹¹ Ibid, hal 213

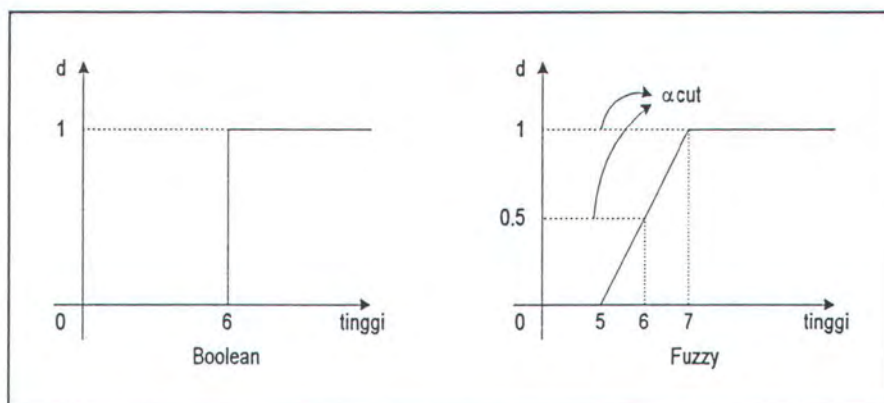
BAB III

TEORI LOGIKA FUZZY

3.1 Pendahuluan

Pada pertengahan tahun 1965 Prof. Lotfi Zadeh dari Universitas California di Berkeley memperkenalkan teori logika fuzzy. Teori ini merupakan generalisasi dari logika multi nilai dan logika konvensional atau logika boolean dalam kasus-kasus tertentu. Beberapa tahun kemudian teori ini dikembangkan ke arah aplikasi kontrol praktis.

Fungsi utama dari logika fuzzy adalah untuk aplikasi kontrol dengan mendefinisikan term dan rule yang intuitif sebagai pengganti fungsi matematis yang kompleks atau tidak linear. Dengan demikian logika fuzzy merupakan aproksimasi dari penalaran manusia. Perbedaan utama antara logika fuzzy dan logika konvensional adalah logika fuzzy tidak hanya mengevaluasi dua nilai, 'true' atau 'false', tetapi lebih dari itu fuzzy memberikan/mengijinkan derajat keanggotaan dari beberapa set/himpunan serta memungkinkan range yang kontinu.



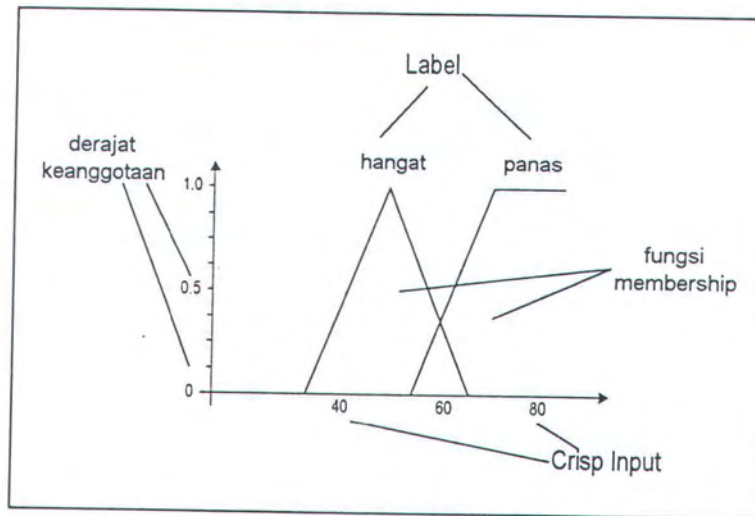
Gambar 3.1¹¹

Fungsi Membership Boolean dan Fuzzy

¹¹ —, FUZZY MICROCONTROLLER DEVELOPMENT SYSTEM, American Neuralogix, Inc., 1992, p. 5-1.

Sebagai contoh klasik, suatu elemen secara pasti hanya mempunyai dua kemungkinan, menjadi anggota atau tidak. Tetapi dalam fuzzy elemen itu dapat mempunyai kemungkinan menjadi anggota dari beberapa set/himpunan dengan nilai keanggotaan (*degree of membership*) yang terletak antara 0 dan 1, seperti terlihat pada gambar 3.1.

Bagian-bagian dari fungsi membership.

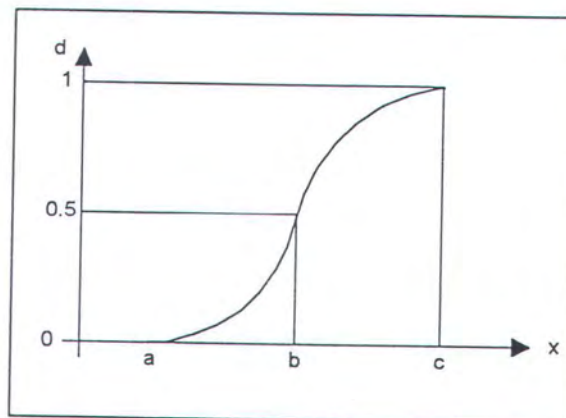


Gambar 3.2

Istilah-istilah dalam Fungsi Membership

Macam-macam bentuk fungsi membership:

◆ Fungsi S



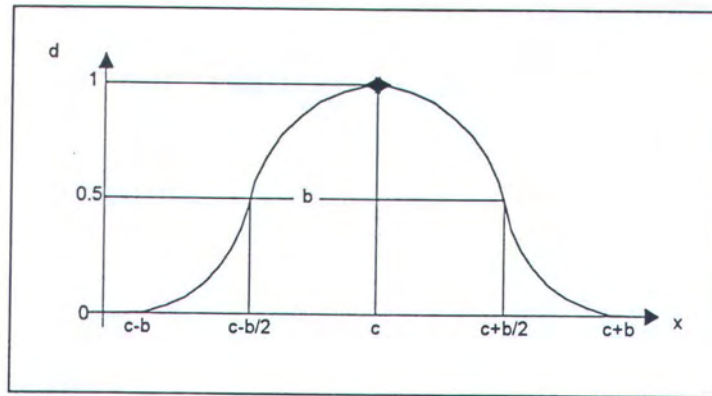
Gambar 3.3

Fungsi S

atau secara matematis:

$$S(d; a, b, c) = \begin{cases} 0 & \text{untuk } d < a \\ 2[(d-a)/(c-a)]^2 & \text{untuk } a \leq d \leq b \\ 1 - 2[(d-c)/(c-a)]^2 & \text{untuk } b \leq d \leq c \\ 1 & \text{untuk } d > c \end{cases}$$

◆ Fungsi π



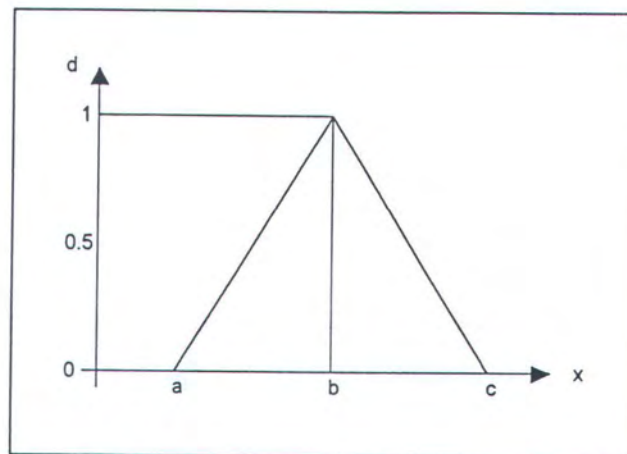
Gambar 3.4

Fungsi π

atau secara matematis:

$$\pi(d; a, b, c) = \begin{cases} S(d; c-b, c-b/2, c) & \text{untuk } d \leq c \\ 1 - S(d; c, c+b/2, c+b) & \text{untuk } d \geq c \end{cases}$$

◆ Fungsi T



Gambar 3.5

Fungsi T

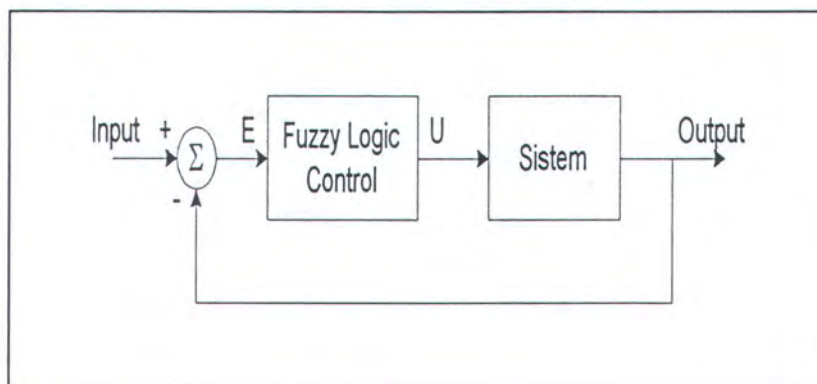
atau secara matematis:

$$S(d; a, b, c) = \begin{cases} 0 & \text{untuk } d < a \\ (d - a)/(b - a) & \text{untuk } a \leq d \leq b \\ (c - d)/(c - b) & \text{untuk } b \leq d \leq c \\ 1 & \text{untuk } d > c \end{cases}$$

3.2 Struktur Dasar Logika Fuzzy

Fuzzy logic controller secara tipikal dapat digolongkan ke dalam sistem kontrol close-loop seperti pada gambar 3.6. Pada gambar 3.7 diperlihatkan elemen utama dari fuzzy logic controller adalah unit fuzzifikasi, unit penalaran fuzzy, data dasar pengambilan keputusan fuzzy (knowledge base), dan unit defuzzifikasi.

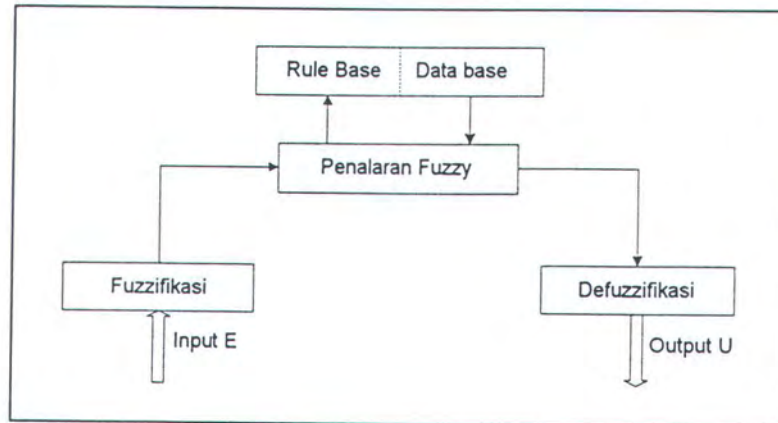
Unit *data base* (*knowledge base*) terdiri dari dua bagian utama, yaitu *data base* untuk mendefinisikan fungsi membership dan *rule base* yang menghubungkan nilai fuzzy input dengan nilai fuzzy output.



Gambar 3.6¹²

Tipikal Sistem Kontrol Dengan Fuzzy Logic

¹² Yan, Jun, Ryan, Michael, Power, James, USING FUZZY LOGIC, Prentice Hall, Ltd., 1994, hal. 46

Gambar 3.7¹³

Struktur Dasar Fuzzy Logic Control

3.2.1 Unit Fuzzifikasi

Fuzzifikasi adalah proses memetakan (*mapping*) crisp input ke dalam set/himpunan fuzzy. Data crisp yang sudah ter-map diubah menjadi variabel label dari fungsi membership yang sesuai (nilai fuzzy input). Definisi fuzzifikasi:

$$x = \text{fuzzifier}(x_o)$$

dimana : x_o : crisp input

x : set/himpunan fuzzy

fuzzifier : fuzzifikasi yang memetakan crisp input ke dalam set fuzzy

3.2.2 Unit Data Dasar Pengambilan Keputusan Fuzzy (Knowledge Base)

Knowledge base terdiri dari *data base* dan *rule base*. *Data base* merupakan pendefinisian dari parameter-parameter fuzzy sebagai set/himpunan fuzzy atau mendefinisikan fungsi membership dari tiap-tiap range variabel. Dalam mendefinisikan *data base* terdapat

¹³ Ibid, hal. 47

beberapa pertimbangan, yaitu range (*universe of discourse*) tiap-tiap variabel, jumlah set/himpunan fuzzy, dan bentuk fungsi membership yang digunakan.

Rule base mengandung pendefinisian *rule control fuzzy* untuk mengatur kerja sistem, sehingga diperoleh sistem kontrol yang diinginkan. *Rule base* ini mencerminkan penalaran manusia terhadap sistem kontrol tersebut. Pendefinisian rule secara garis besar adalah sebagai berikut:

Rule 1 IF x_1 is A_{11} AND ... AND x_m is A_{1m} THEN y is B_1

Rule 2 IF x_1 is A_{21} AND ... AND x_m is A_{2m} THEN y is B_2

Rule n IF x_1 is A_{n1} AND ... AND x_m is A_{nm} THEN y is B_n

3.2.3 Unit Defuzzifikasi

Defuzzifikasi adalah proses memetakan (*mapping*) nilai output fuzzy ke nilai *non-fuzzy* (crisp) dan dapat dinyatakan sebagai berikut:

$$y_o = \text{defuzzifier}(y)$$

dimana y : nilai output fuzzy

y_o : nilai *non-fuzzy (crisp)* dan

defuzzifier : proses defuzzifikasi yang didefinisikan pemrogram

Pada umumnya metode defuzzifikasi yang digunakan adalah:

- ♦ Metode mean of maximum (MOM).

Metode, juga disebut *height defuzzification*, adalah metode yang menghasilkan nilai output rata-rata dari nilai-nilai output maximum fungsi membership.

- ♦ Metode center of gravity (COG).

Metode ini menghasilkan nilai output yang merupakan gravity dari distribusi nilai output fungsi membership. Metode ini paling banyak digunakan.

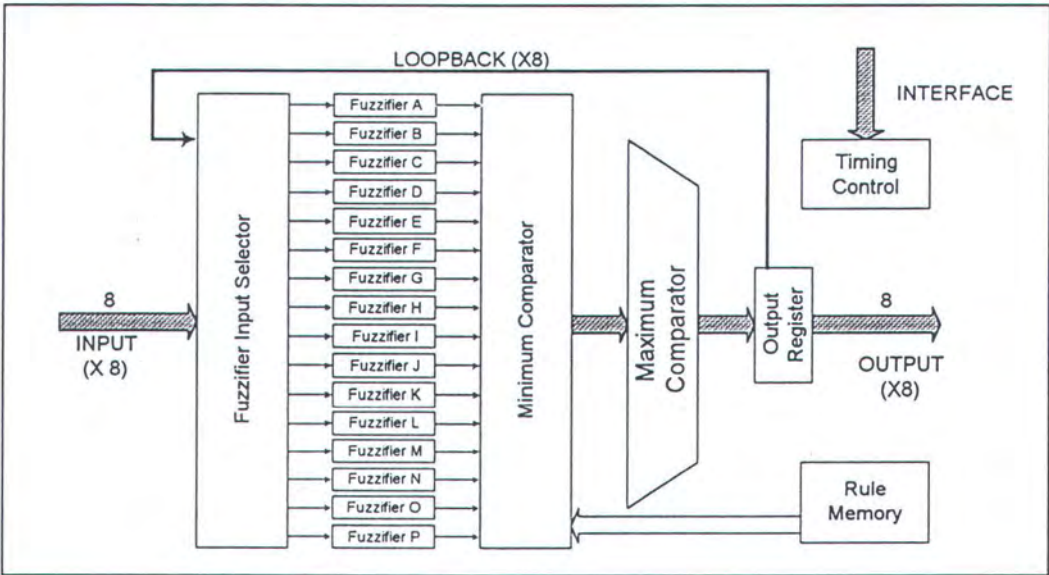
3.3 Chip Fuzzy NLX230

3.3.1 Pendahuluan

Chip NLX230 merupakan mikrokontroller yang menerapkan logika fuzzy untuk menghasilkan output (*action value*) berdasarkan input serta fungsi membership yang telah didefinisikan oleh pemrogram. Untuk keperluan efisiensi implementasi digital, chip ini menerapkan fungsi membership linier, berbentuk T dan simetris serta menggunakan metode pengambilan keputusan/penalaran fuzzy (*fuzzy inference*) yang sederhana yaitu metode **MIN-MAX**

Rule berfungsi untuk menentukan kondisi-kondisi input yang dimungkinkan. Rule didefinisikan dengan memetakan crisp input dengan fungsi membership dan nilai aksi (*action value*) yang ditentukan oleh pemrogram. Rule yang sesuai dengan kondisi input menentukan perubahan pada nilai output. Jumlah rule yang dapat dibuat maksimal 64 rule.

3.3.2 Deskripsi Rangkaian Dalam NLX230



Gambar 3. 8¹⁴

Blok diagram NLX230

¹⁴ Op Cit, p. 6-2

3.3.2.1 Fuzzifier

Chip ini mempunyai 16 *fuzzifier*. Tiap *fuzzifier* memproses hubungan crisp input dengan fungsi membership yang telah didefinisikan dalam *term*. Jumlah term yang dapat didefinisikan untuk chip ini adalah 16 term. Tiap *fuzzifier* menentukan seberapa jauh jarak nilai input terhadap nilai pusat (*center value*). Nilai pusat merupakan nilai di mana input benar-benar berada di dalam fungsi membership yang bersangkutan (logika 1). Hasil perhitungan jarak ini kemudian dibandingkan dengan *width* yang telah didefinisikan oleh pemrogram. Jika nilai jarak kurang atau sama dengan *width*, maka input merupakan bagian dari fungsi membership dan selanjutnya nilai jarak ini dikomplemenkan untuk mendapatkan derajat keanggotaan dari fungsi membership yang bersangkutan.

Yang dimaksud dengan mengkomplemenkan adalah mengurangi nilai jarak terhadap nilai 30. Semakin dekat input ke nilai pusat semakin tinggi nilai membership. Nilai membership ini kemudian dimasukkan ke minimum komparator untuk proses rule.

3.3.2.2 Minimum Comparator

Bagian ini mempunyai 24 bit rule dengan 16 bit pertama digunakan untuk meng-enable *output fuzzifier*. Tiap rule bit berhubungan secara pasti (*fix*) dengan satu *fuzzifier*.

Pada saat rule bit dimasukkan (*asserted* = 1), maka tiap-tiap *output fuzzifier* dibandingkan satu sama lain untuk mendapatkan nilai minimum.

3.3.2.3 Maximum Comparator

Nilai minimum disimpan ke dalam maximum register di maximum komparator dan kemudian dibandingkan dengan nilai referensi yang tersimpan dalam maximum register.

Action Value adalah 8 bit 2's complements dengan range -128 sampai dengan 127.

3.3.3 Mode operasi NLX230

3.3.3.1 Mode Fuzzifier

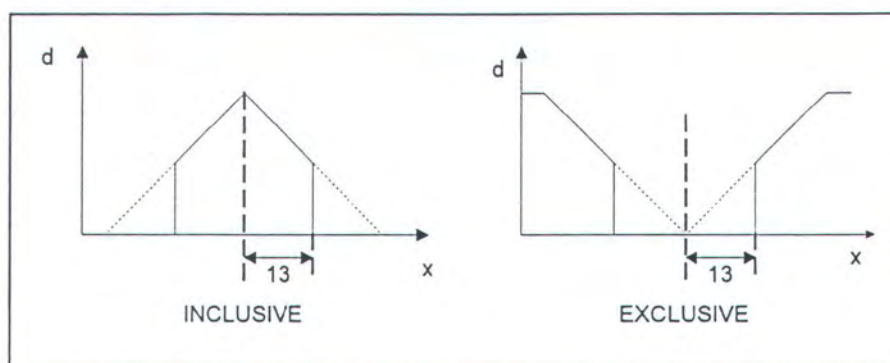
Ada dua cara fuzzifier digunakan, yaitu :

♦ *Inclusive membership,*

Output dari fuzzifier menunjukkan nilai terdekat dari input terhadap lokasi center membership yang bersangkutan.

♦ *Exclusive membership,*

Output dari fuzzifier menunjukkan nilai seberapa jauh input terhadap lokasi center membership yang bersangkutan.



Gambar 3.9¹⁵

Fungsi inclusive dan exclusive

3.3.3.2 Mode Input

Input FMC 8 pin input termultiplex atau internal, loop back, input berasal dari output.

3.3.3.3 Mode Output

Konfigurasi output FMC terdiri dari dua macam, yaitu :

♦ Mode output *immediate,*

¹⁵ Ibid p. 5-10

Action value yang dispesifikasikan untuk rule akan ditambahkan terhadap nilai awal. Pada mode ini nilai awal selalu digunakan untuk perhitungan output tiap siklus proses FMC.

- ◆ Mode output *accumulate*,
Action value ditambahkan terhadap nilai output sebelumnya. Pada mode ini nilai output yang baru merupakan nilai awal untuk perhitungan output berikutnya. Dengan demikian nilai awal sebenarnya hanya digunakan pada saat perhitungan output pertama setelah reset.

3.3.4 Register-register internal FMC

- ◆ Register mode output
Register mode output menentukan tipe dari 8 output, termasuk *mode immediate* atau *accumulate*. Bit yang diset ke logika 0 menunjukkan *mode immediate* dan bit yang diset ke logika 1 menunjukkan *mode accumulate*.

Alamat offset 0C hex (bit ke-96 - 103)

| | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| out 7 | out 6 | out 5 | out 4 | out 3 | out 2 | out 1 | out 0 |

0 = nilai awal + aksi
1 = nilai sebelumnya + aksi

- ◆ Register konfigurasi input
Input ke fuzzifier dapat berasal dari eksternal atau internal loopback. Jika bit I/O register diset 1, maka input berasal dari eksternal. Demikian sebaliknya.

Alamat offset 0D hex (bit ke-104 - 111)

| | | | | | | | |
|---------|---------|---------|---------|---------|---------|---------|---------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| In 7 | In 6 | In 5 | In 4 | In 3 | In 2 | In 1 | In 0 |

0 = output di-loopback

1 = input eksternal

◆ Register tipe fungsi membership (0 dan 1)

Register ini berfungsi untuk menentukan tipe fungsi membership, tipe *inclusive* atau tipe *exclusive*.

Alamat offset 0E hex - 0F hex (bit ke-112 - 127).

| | | | | | | | |
|-------------|-------------|-------------|-------------|-------------|-------------|------------|------------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Fuzzy 7 | Fuzzy 6 | Fuzzy 5 | Fuzzy 4 | Fuzzy 3 | Fuzzy 2 | Fuzzy 1 | Fuzzy 0 |
| Fuzzy 15 | Fuzzy 14 | Fuzzy 13 | Fuzzy 12 | Fuzzy 11 | Fuzzy 10 | Fuzzy 9 | Fuzzy 8 |

0 = Inclusive membership.

1 = Exclusive membership.

◆ Register nilai awal (0 -7)

Setiap output mempunyai nilai awal (range -128 sampai +127). Nilai awal berkaitan dengan mode output seperti dijelaskan sebelumnya.

Alamat offset 10 hex - 17 hex (bit ke-128 - 191).

| | | | | | | | |
|------|-----|---|---|---|---|---|-----|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Sign | MSB | | | | | | LSB |

Bit ke-8 di atas menunjukkan tanda (sign) dari tiap-tiap output.

- ◆ Register rule boundary (0 - 7)

Alamat offset 18 hex - 1F hex (bit ke-192 - 255).

| | | | | | | | |
|---------|---------|-----|---|---|---|---|-----|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Logic 0 | Logic 0 | MSB | | | | | LSB |

Lima bit di atas menunjukkan jumlah rule terakhir yang bersesuaian dengan jumlah output.

- ◆ Register term (0 -15)

Register ini berfungsi untuk mendefinisikan term yang terdiri dari nilai pusat (center value), seleksi input, dan width serta untuk mendefinisikan adanya loopback internal.

Alamat offset 20 hex - 3F hex (bit ke-256 - 511).

| | | | | | | | |
|---------------|-------------|-------------|--------------|---|---|---|---------------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Center MSB | | | | | | | Center LSB |
| Select 2 | Select 1 | Select 0 | Width MSB | | | | Width LSB |

- ◆ Register action value (0 - 63)

Nilai aksi menggunakan metode komplemen 2.

Alamat offset 40 hex - 7F hex (bit ke-512 - 1023)

| | | | | | | | |
|------|-----|---|---|---|---|---|-----|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Sign | MSB | | | | | | LSB |

- ◆ Register rule (0 - 63)

Alamat offset 80 hex - BF hex (bit ke-1024 - 1535)

[illegible]

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|

0 = exclude term

1 = include term

- ◆ Register rule (0 - 63)

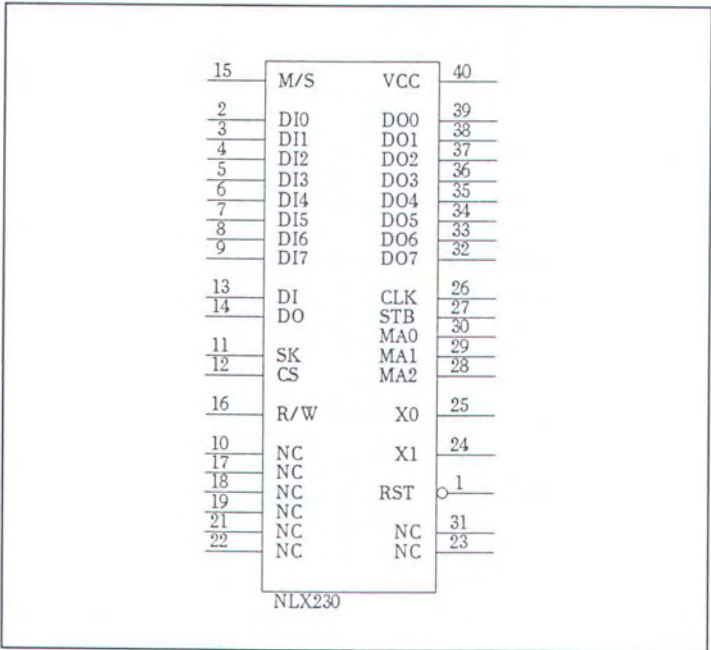
Alamat offset C0 hex - FF hex (bit ke-1536 - 2047)

| | | | | | | | |
|-------------|-------------|-------------|-------------|-------------|-------------|------------|------------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Fuzzy 15 | Fuzzy 14 | Fuzzy 13 | Fuzzy 12 | Fuzzy 11 | Fuzzy 10 | Fuzzy 9 | Fuzzy 8 |

0 = exclude term

1 = include term

3.3.5 Deskripsi Pin



Gambar 3.10
Pin-pin NLX230

◆ DI (7:0) : *Data Input*

Data merupakan TDM (*Time Division Multiplexed*) 8 bit byte berdasarkan MA (2:0) dan STB.

◆ DO(7:0) : *Data Output*

Data merupakan TDM (*Time Division Multiplexed*) 8 bit byte berdasarkan MA (2:0) dan STB.

◆ STB : *Strobe*

Digunakan bersama-sama dengan MA untuk mendekode channel input dan output yang sedang aktif.

◆ MA (2:0) : *MUX Address*

Tiga bit output untuk mendekode channel input dan output yang sedang aktif.

◆ SK : *Serial Clock*

Bagian dari konfigurasi interface. Pada mode master SK digunakan meng-outputkan clock ke eksternal EEPROM. Pada mode ini frekuensi SK adalah input clock dibagi 32 ($X1/32$).

Pada *mode slave* SK berfungsi sebagai input clock yang berasal dari host komputer pada proses *download*.

◆ CS : *Chip Select*

Bagian dari konfigurasi interface. Pada *mode master* CS berfungsi mengeluarkan sinyal *select* ke eksternal EEPROM.

Pada *mode slave* CS berfungsi sebagai input (aktif low) yang merupakan fungsi dari dekoder alamat selama proses *download*.

◆ *DI : Serial Data Input*

Bagian dari konfigurasi interface. Pada *mode master* DI berfungsi sebagai input data serial dari eksternal EEPROM.

Pada *mode slave* DI berfungsi sebagai input data serial dari *host* komputer.

◆ *DO : Serial Data Output*

Bagian dari konfigurasi interface. Pada *mode master* DO berfungsi untuk mengeluarkan sinyal kontrol (*read*) dan alamat ke pin data input eksternal EEPROM.

Pada *mode slave* DO berfungsi untuk mengeluarkan konfigurasi memori internal chip ke *host* komputer.

◆ *R/W : Read-Write*

Bagian dari konfigurasi interface. Pada *mode master* pin ini tidak berfungsi dan harus di-*pull up*.

Pada *mode slave* R/W (logika 1) berfungsi mengeluarkan sinyal *read*, yaitu membaca konfigurasi memori internal chip dan (logika 0) mengeluarkan sinyal *write*, yaitu memasukkan konfigurasi memori internal chip.

◆ *XI : Crystal Oscillator Input*

Digunakan berpasangan dengan XO. Pin ini dapat dikemudikan langsung oleh TTL untuk meng-input-kan eksternal clock.

◆ *XO : Crystal Oscillator Output*

Digunakan berpasangan dengan XI. Jika digunakan Eksternal clock pin ini tidak perlu dihubungkan.

◆ M/S : *Master/Slave Mode Control*

Bagian dari konfigurasi interface. Pada *mode master* (logika 1) chip mengeluarkan sinyal-sinyal kontrol untuk membaca EEPROM segera setelah negasi dari RST (transisi dari low ke high). Pin SK, CS, dan DO menjadi output, mengeluarkan sinyal-sinyal untuk membaca EEPROM melalui pin DI.

Pada *mode slave* chip menunggu sinyal SK, CS, RW, dan DI dari host komputer.



BAB IV
PERANCANGAN PERANGKAT KERAS
DAN PERANGKAT LUNAK

BAB IV

PERANCANGAN PERANGKAT KERAS DAN PERANGKAT LUNAK

4.1 Permasalahan

Pada dasarnya alat kontrol kualitas air minum dan air baku ini diinginkan agar memiliki fungsi-fungsi sebagai berikut:

- ♦ peralatan dapat mengambil data derajat keasaman (pH), kadar oksigen terlarut dalam air, dan tingkat kejernihan.
- ♦ peralatan dapat menjalankan fungsi kontrol kualitas air, sehingga akan didapat kondisi kualitas air yang optimum, baik untuk air baku yang akan diolah menjadi air minum atau pun air minum yang siap didistribusikan ke pelanggan.
- ♦ penyajian grafik parameter-parameter terukur ke layar monitor.

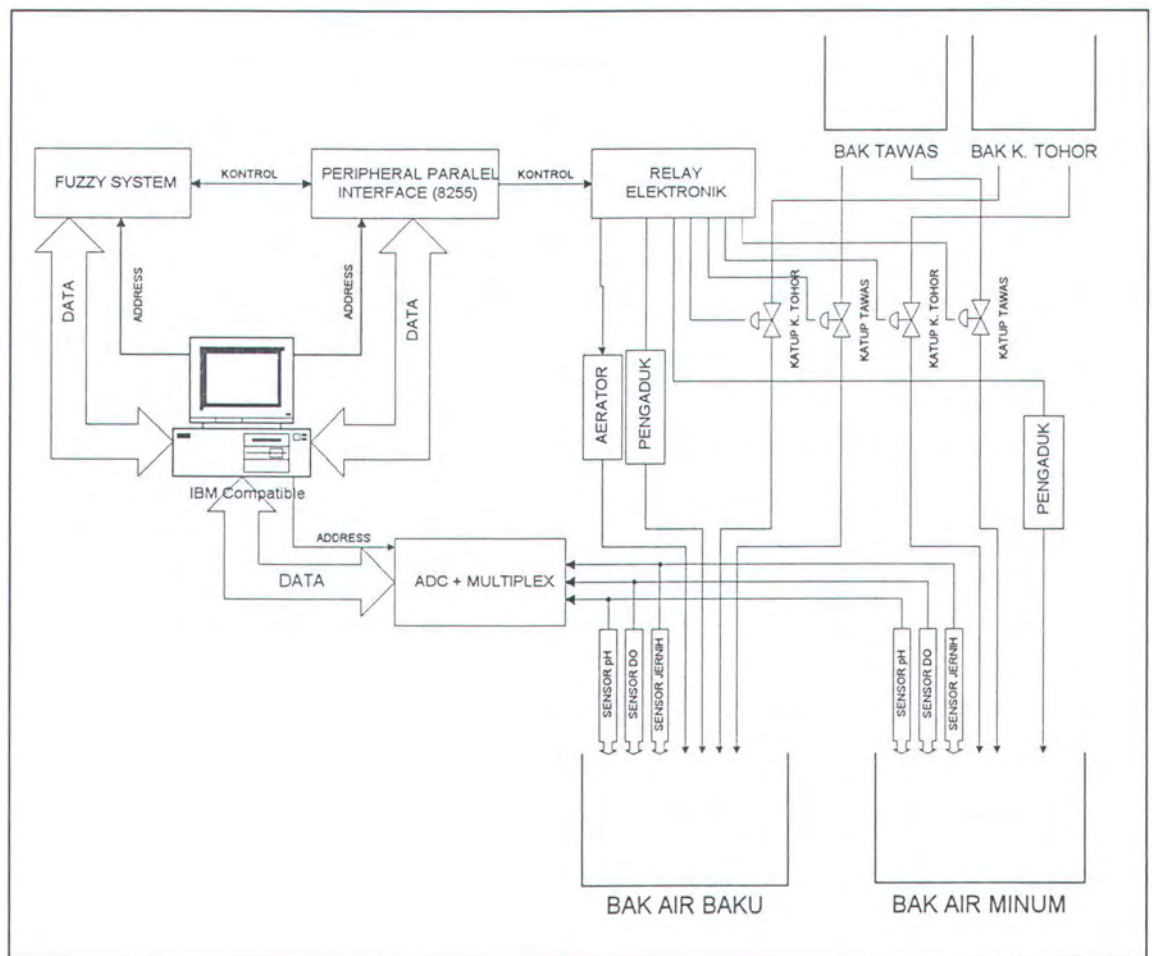
4.2 Desain Perangkat Keras

Dengan fungsi sebagaimana yang diinginkan, perangkat keras Tugas Akhir ini memerlukan sub-sistem dasar:

- ♦ IBM PC.
- ♦ Fuzzy card berbasis chip fuzzy NLX230.
- ♦ Antar-muka paralel 8255.
- ♦ Analog to Digital Converter.
- ♦ Instrumentasi transduser pH.

- ◆ Instrumentasi transduser DO.
- ◆ Instrumentasi fotodiode sebagai transduser tingkat kejernihan.
- ◆ Kapasitansi meter.
- ◆ Driver katup/solenoid valve.

Hubungan antar sub-sistem tersebut dapat diilustrasikan pada gambar 4.1.



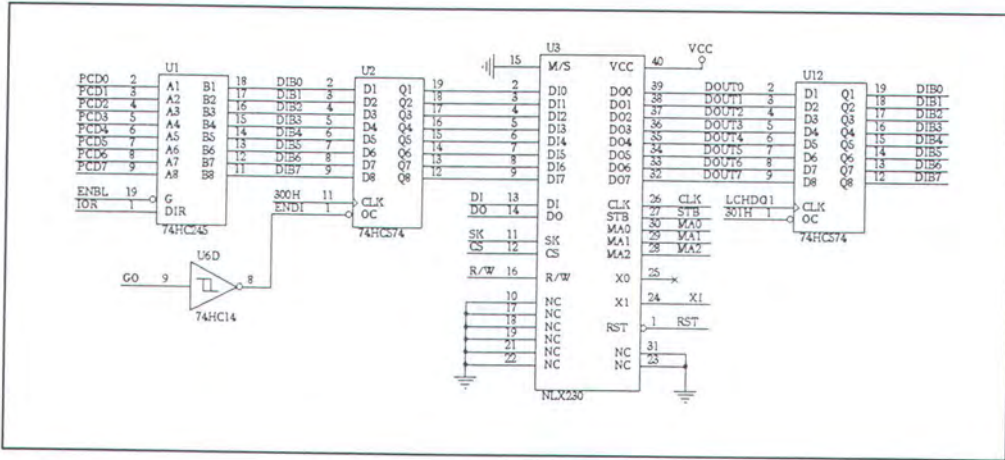
Gambar 4.1
Blok Diagram Sistem

Sub sistem IBM PC memiliki fungsi sebagai pengendali sistem rangkaian yang meliputi pengambilan data, pengolahan data dengan logika fuzzy dan penyajiannya ke monitor serta pengatur prosedur pengetesan.

4.2.1 Modul Fuzzy NLX230

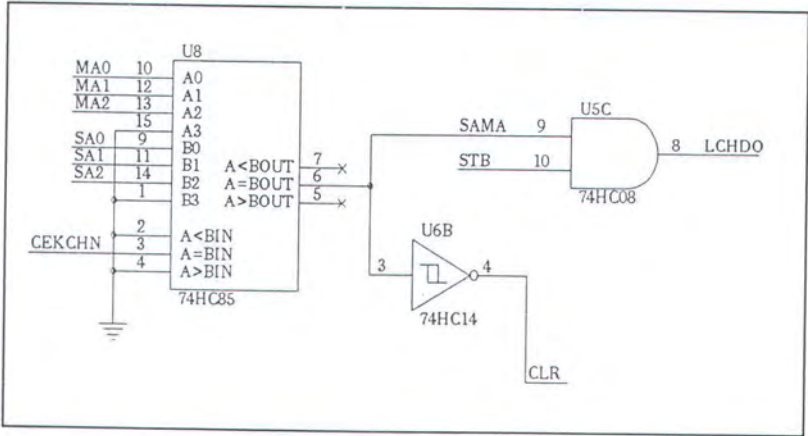
Modul fuzzy ini dikontrol oleh antar-muka paralel 8255, yang meliputi kontrol download dan upload, kontrol input dan output data, serta kontrol proses running NLX230.

Chip NLX230 dikonfigurasi dengan buffer-buffer data. Rangkaiannya adalah sebagai berikut:



Gambar 4.2
Buffer data NLX230

NLX230 mempunyai delapan channel sebagai pemroses data yang bekerja secara paralel. Untuk memilih channel-channel tersebut harus dibuatkan sebuah komparator channel. Rangkaian komparator channel adalah sebagai berikut:



Gambar 4.3
Komparator channel

4.2.2 Antar-muka Paralel

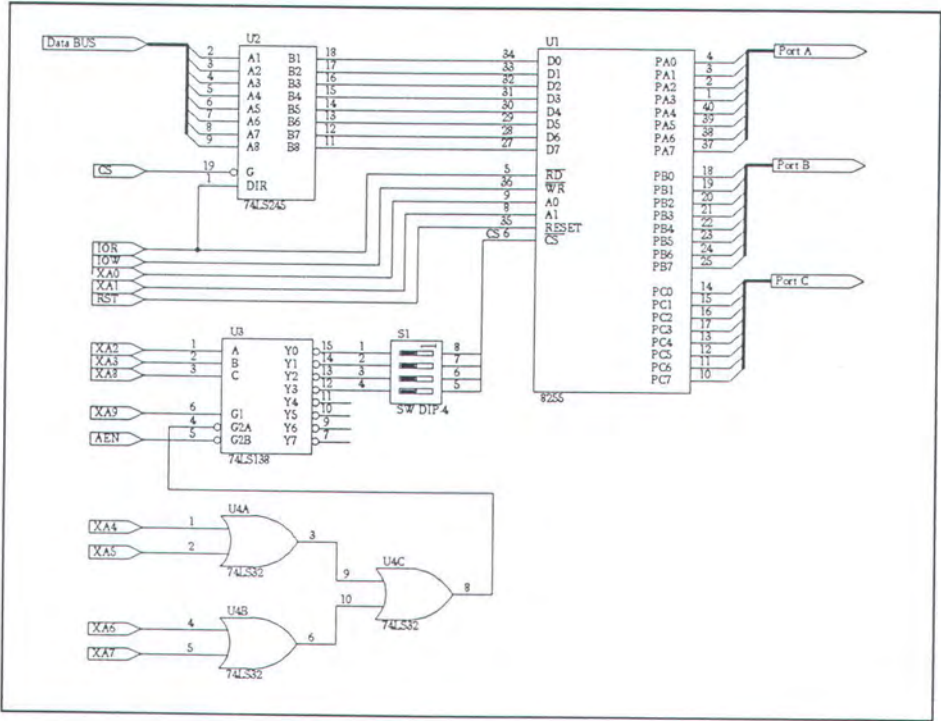
Antar-muka paralel pada tugas akhir ini menggunakan Programmable Peripheral Interface 8255 dengan address port 308h - 30Bh, berfungsi untuk mengendalikan bit-bit kontrol:

- ◆ proses kontrol download, upload, dan running NLX230.
- ◆ selektor multiplexer analog.
- ◆ selektor relay katup dan motor pengaduk.

Address port yang dipergunakan adalah:

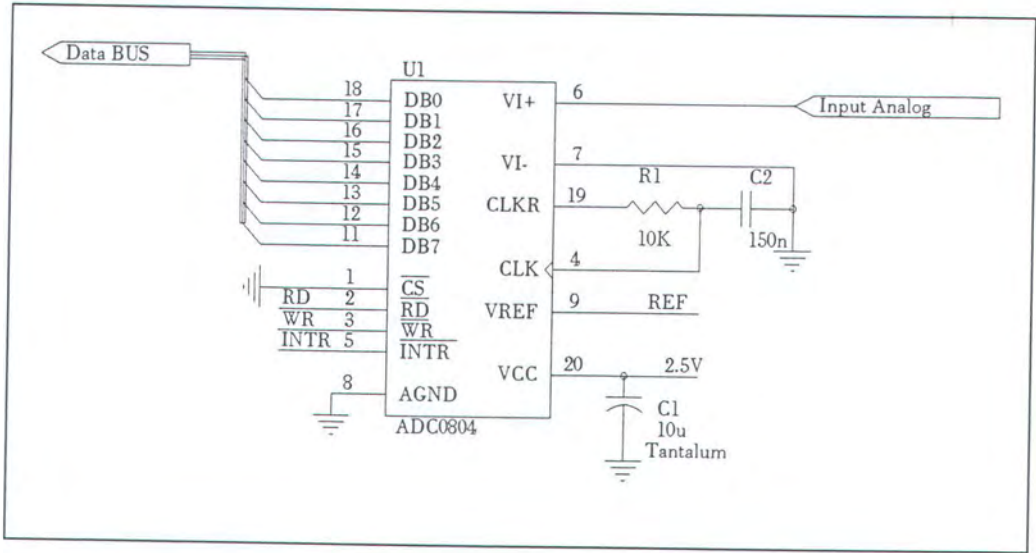
- ◆ Port A 308h.
- ◆ Port B 309h.
- ◆ Port C 30Ah.
- ◆ Port Control Word 30Bh.

Antar muka-paralel ini dikonfigurasi; port A , port B, dan port C lower sebagai output serta port C upper sebagai input. Rangkaian antar-muka dapat dilihat pada gambar 4.4



Gambar 4.4
Rangkaian antar-muka paralel

Adapun rangkaian ADC0804 adalah sebagai berikut:

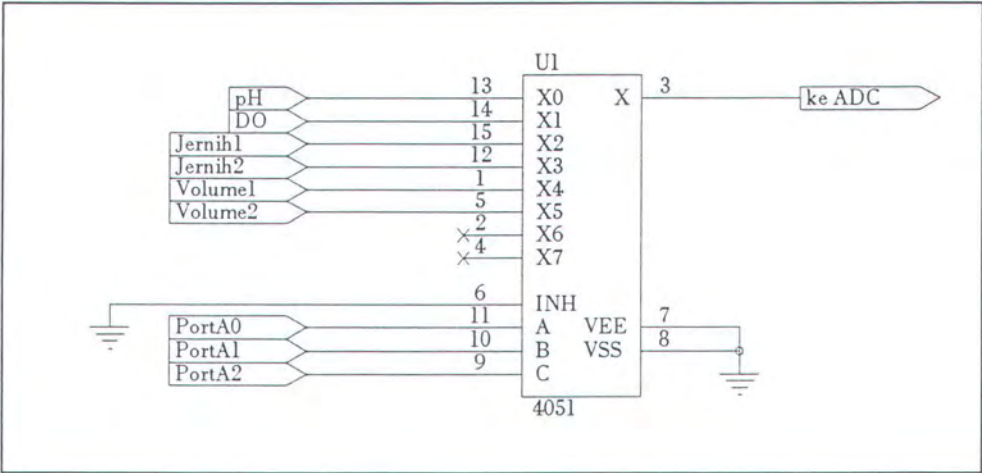


Gambar 4.6
Rangkaian ADC0804

Rangkaian tegangan referensi yang digunakan untuk input tegangan referensi ADC ini dihasilkan oleh sebuah dioda referensi presisi 2,5 volt jenis IC LM 335 - 2,5 volt untuk menghasilkan output ADC dengan resolusi 20mV / bit.

4.2.4 Multiplekser Analog

Multiplexer analog dipergunakan untuk memilih salah satu input yang akan dihubungkan ke ADC. Komponen aktif yang digunakan adalah IC CMOS 4051 yaitu multiplekser analog 8 ke 1. Pemilihan channel aktif dilakukan oleh port antar-muka paralel sebanyak 3 bit. Selengkapnya rangkaian multiplexer analog dapat dilihat di bawah ini :



Gambar 4.7
Multiplekser Analog

Tabel channel yang aktif untuk proses akuisisi data adalah sebagai berikut:

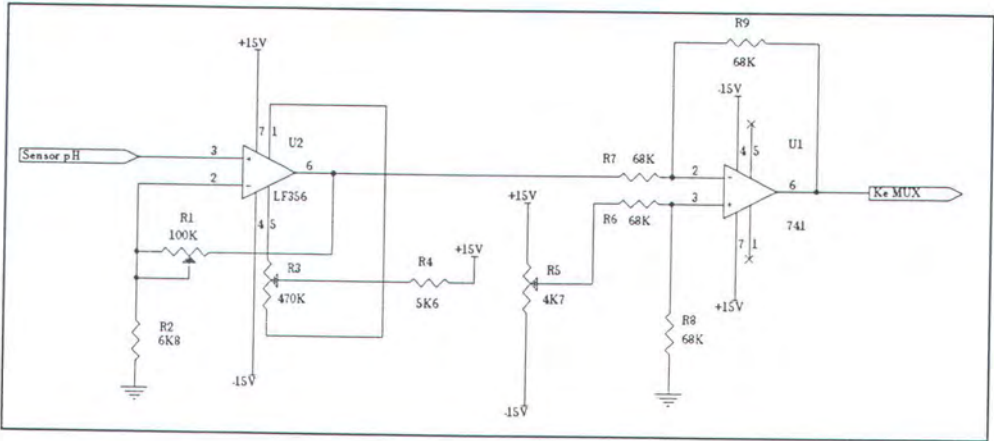
Tabel 4.1
Channel yang aktif

| Port A2 | Port A1 | Port A0 | Channel yang Aktif |
|---------|---------|---------|--------------------|
| 0 | 0 | 0 | pH |
| 0 | 0 | 1 | DO |
| 0 | 1 | 0 | Jernih1 |
| 0 | 1 | 1 | Jernih2 |

4.2.5 Instrumentasi pH

Instrumentasi transducer pH untuk mendeteksi derajat asam-basa larutan menggunakan komponen aktif op-amp yang dikonfigurasi sebagai penguat non inverting dan sebagai penguat diferensial.

Rangkaiannya adalah sebagai berikut :

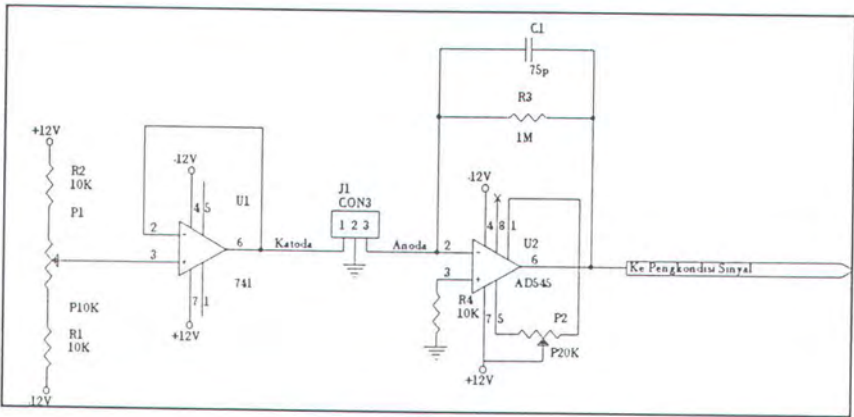


Gambar 4.8
Instrumentasi pH

4.2.6 Instrumentasi Kadar Oksigen Terlarut

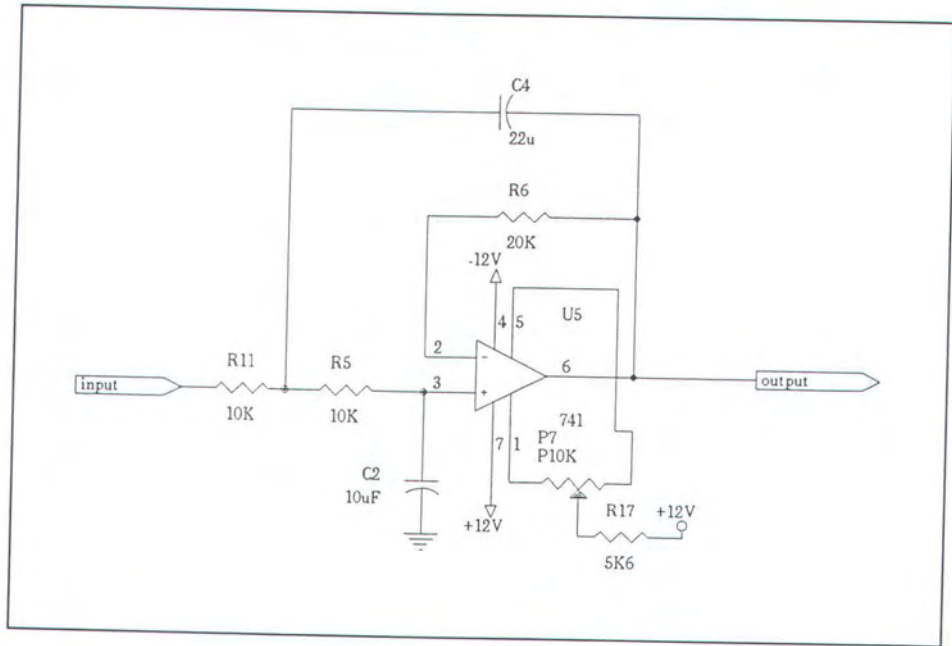
Untuk mengukur kadar oksigen terlarut digunakan transduser dengan tipe HI 76401 yang menghasilkan output berupa arus dengan range 0,1 nA saat *zero oxigen* dan 100 nA saat *oxigen saturation*, karena itu sebelum output transduser dikondisi dengan pengkondisi sinyal harus diubah terlebih dahulu ke tegangan dengan rangkaian pengubah arus ke tegangan. Bagian akhir dari instrumentasi DO ini harus difilter *low pass* untuk menghilangkan noise. Filter yang diterapkan di sini adalah *low pass filter* dengan *cut off frekuensi* 1 Hz.

Rangkaian instrumentasi transduser DO:



Gambar 4.9
Instrumentasi DO

Rangkaian filter *low pass* dengan frekuensi *cut off* 1Hz sebagai berikut:



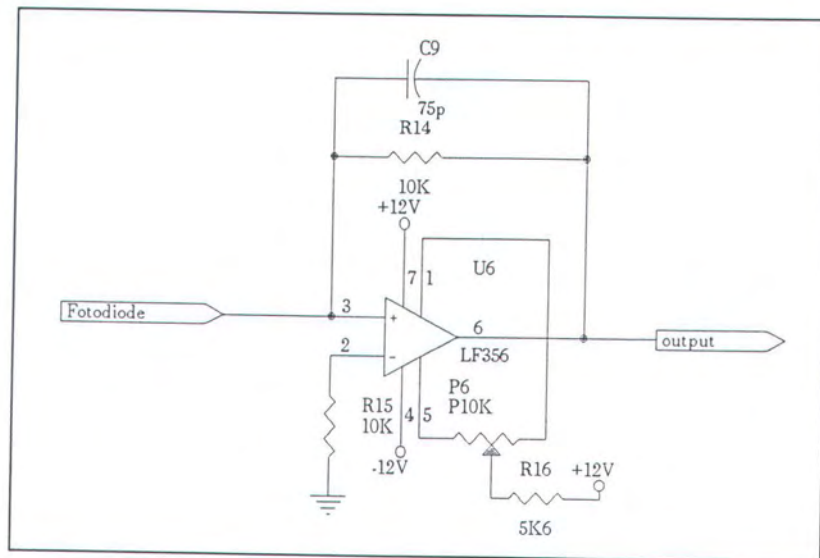
Gambar 4.10

Rangkaian low pass filter

4.2.7 Instrumentasi Kejernihan Air

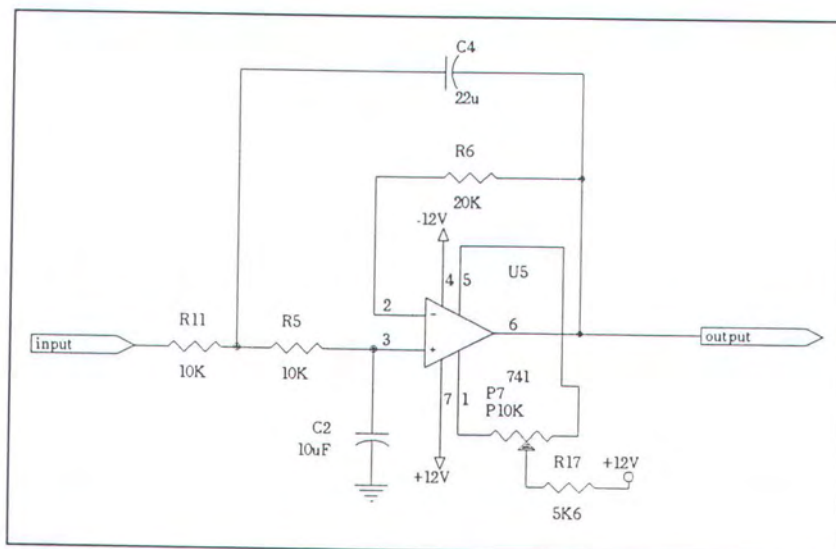
Untuk mendeteksi kejernihan air digunakan fotodiode dan lampu pijar 12 VDC yang dikonfigurasi sedemikian hingga dapat digunakan sebagai alat pendeteksi kejernihan air. Resistansi fotodiode berubah seiring dengan perubahan intensitas cahaya yang mengenainya. Media rambat cahaya adalah air. Dengan demikian kejernihan air mempengaruhi intensitas cahaya yang diterima fotodiode.

Rangkaian instrumentasi kejernihan air.



Gambar 4.11
Instrumentasi kejernihan air

Karena output dari instrumentasi kejernihan air mengandung noise, maka dipasang filter *low pass* dengan *cut off frekuensi* 1 Hz.

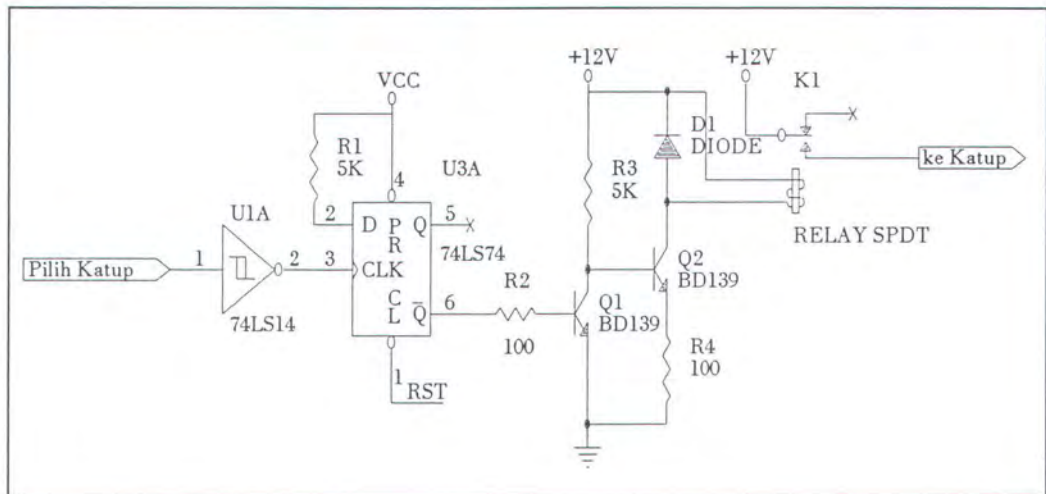


Gambar 4.12
Filter low pass

4.2.8 Driver Katup

Penambahan larutan tawas atau kapur tohor diatur dengan pengaktifan katup untuk membuka. Untuk mengisolasi rangkaian dengan katup digunakan relay. Pengaktifan relay menyebabkan katup juga aktif.

Ada pun rangkaian driver katup adalah sebagai berikut:



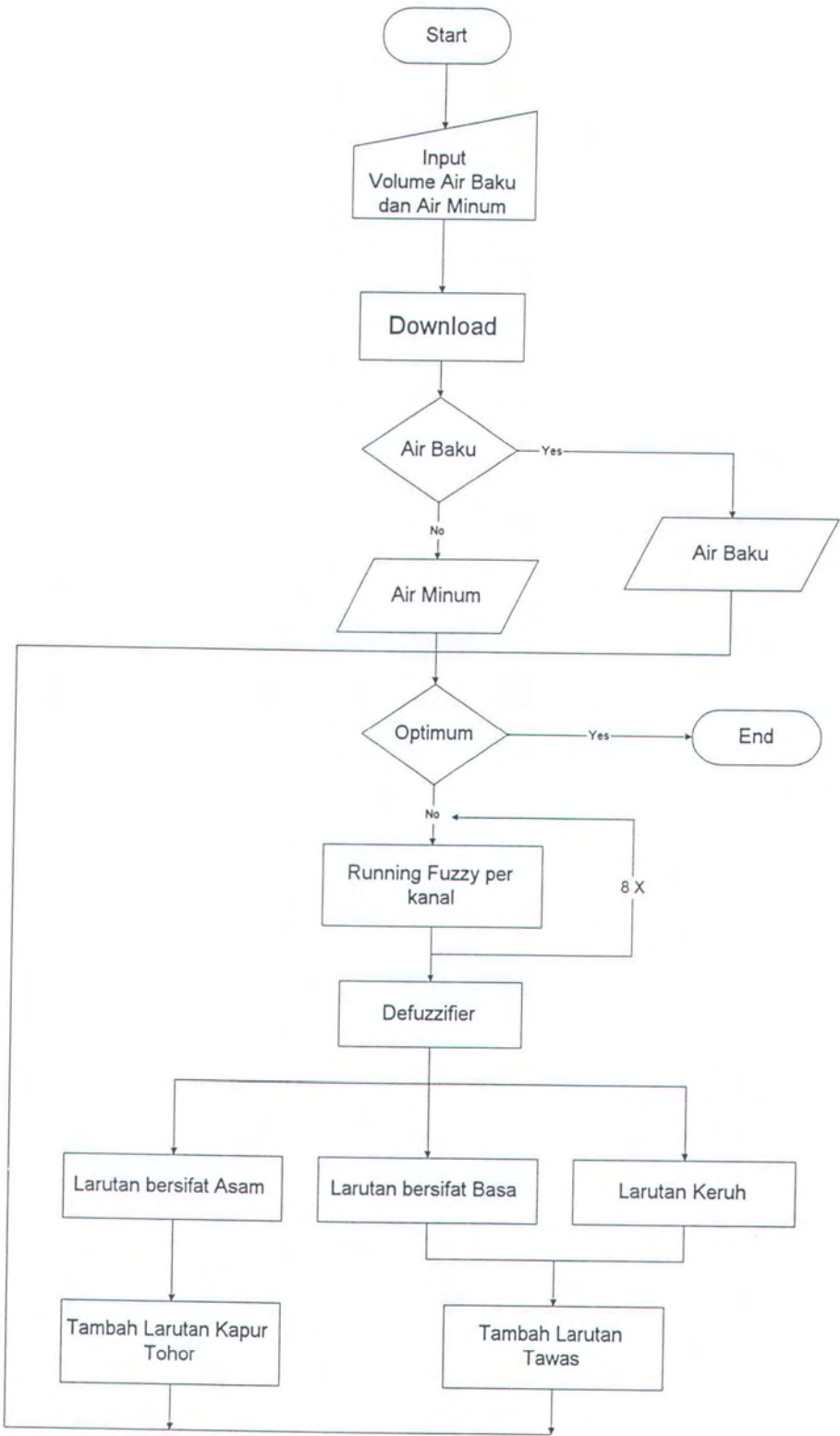
Gambar 4.13

Rangkaian Driver Katup

4.3 Perencanaan Perangkat Lunak

Perencanaan perangkat lunak mencakup dua hal pokok yaitu mengenai proses pengambilan dan pengolahan data yang telah diambil untuk kemudian dipakai sebagai dasar pertimbangan untuk perlu atau tidaknya dilakukan proses pengkondisian air. Bahasa pemrograman yang digunakan adalah bahasa Pascal.

Untuk mempermudah logika pemrograman perlu dibuat diagram alur dari program. Diagram alur itu berisikan urutan logika pemrograman sehingga program dapat dibuat lebih terstruktur dan sesuai dengan kondisi perangkat keras yang telah dibuat. Dengan demikian terdapat satu kesatuan antara perangkat keras dan perangkat lunak. Gambar 4.14 memperlihatkan diagram alur program utama.



Gambar 4.14
Diagram Alir Program

Agar pemrograman menjadi terstruktur, maka secara garis besar program utama terdiri dari beberapa sub program yang berupa prosedur-prosedur program. Sub-sub program tersebut antara lain adalah sub program download dan upload, sub program running NLX230, sub program pengambilan data, sub program pemrosesan data, sub program pengaktifan katup. Masing-masing sub program akan dijelaskan satu-persatu berikut ini:

4.3.1 Bitmap dari term dan rule

Term dan rule yang sudah didefinisikan diisikan ke dalam internal register NLX230 dalam bentuk data biner. Untuk mengubah term dan rule tersebut ke biner dipergunakan *software education FMC development system*. Hasil pengubahan ini merupakan data byte sebanyak internal register yang dimiliki NLX230 yaitu mulai dari address 00h sampai FFh. Term, rule, dan bitmap terlampir.

4.3.2 Download dan Verify

Pemgontrolan pengisian data biner ke dalam internal register NLX230 (*download*) dan untuk mengecek (*verify*) data yang sudah diisikan ke internal register dengan menggunakan bit-bit antar muka-paralel (port C). Bit-bit tersebut adalah sebagai berikut:

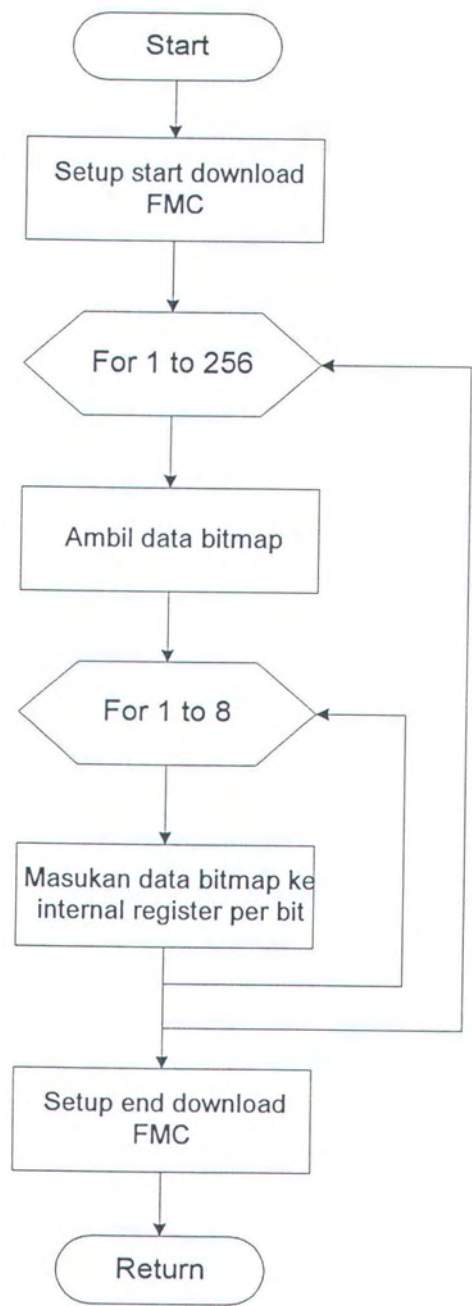
Tabel 4.2
Kontrol Download

| | | | | | | | |
|-----|-----|--------|-----|-----|--------|---------|--------|
| PC7 | PC6 | PC5 | PC4 | PC3 | PC2 | PC1 | PC0 |
| - | - | FMC DO | - | - | FMC DI | FMC R/W | FMC CS |

Tabel 4.3
Kontrol Verify

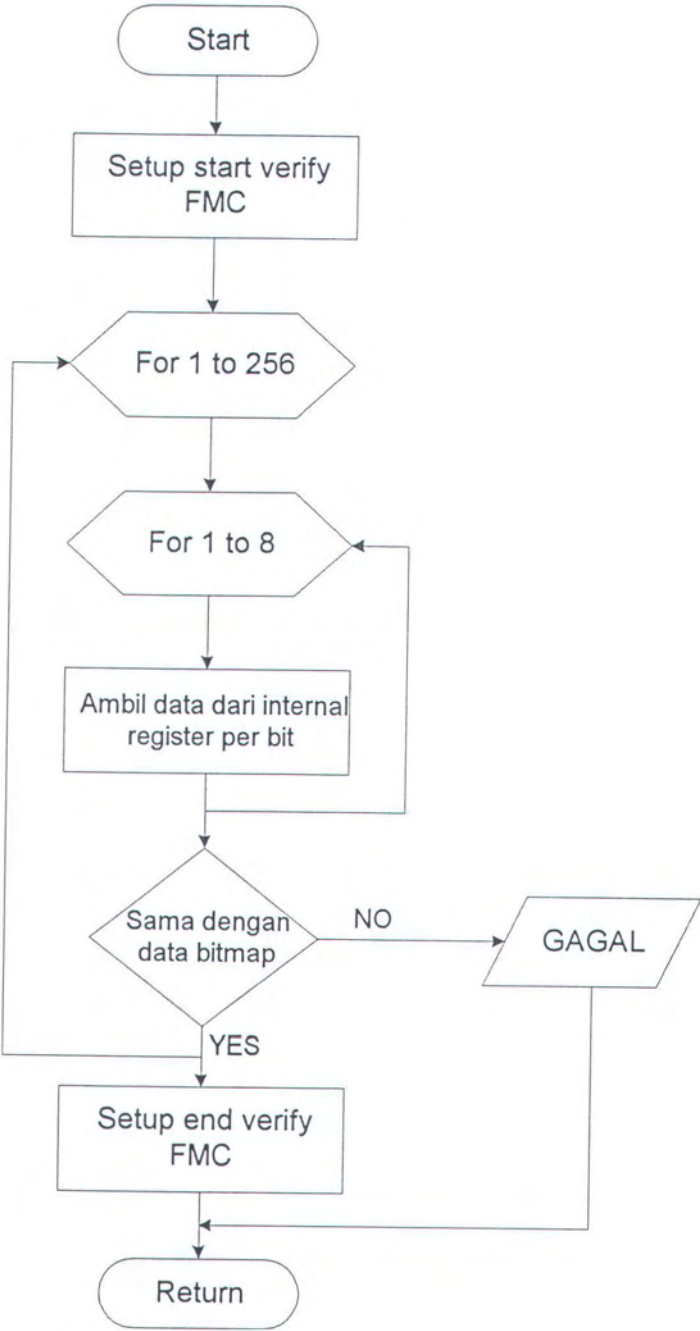
| | | | | | | | |
|-----|-----|----------|---------|-----|-----|-----|-----|
| PB7 | PB6 | PB5 | PB4 | PB3 | PB2 | PB1 | PB0 |
| - | RST | FMC Step | CLK Sel | Go | SA2 | SA1 | SA0 |

Flowchart Download:



Gambar 4.15
Diagram Alir Download

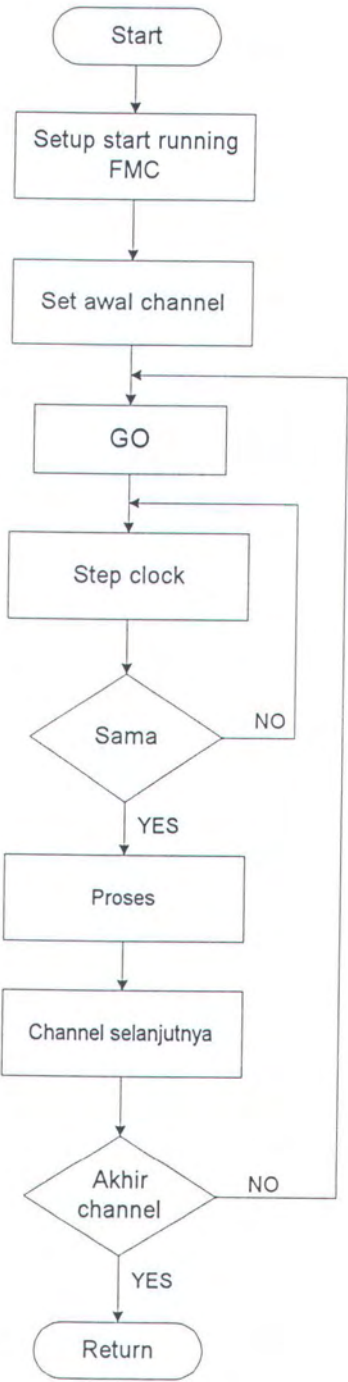
Flowchart verify:



Gambar 4.16
Diagram Alir Verify

4.3.3 Running

Data yang diinputkan ke NLX230 diolah dengan menggunakan logika fuzzy dengan berdasarkan pada data term dan rule yang sudah didownload. Diagram alir proses pengolahan data (running) ini sebagai berikut:



Gambar 4.17
Diagram Alir Running FMC

4.3.4 Defuzzifier

Membership yang digunakan untuk defuzzifier adalah membership tipe singleton.

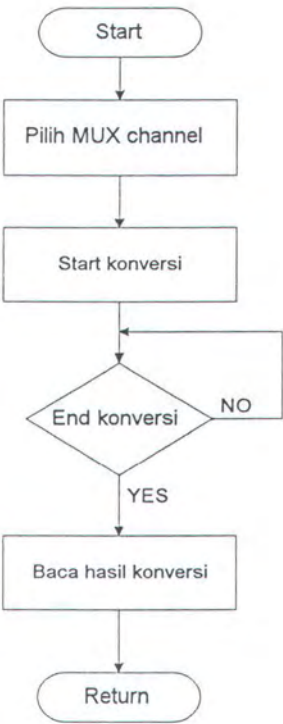
Tabel 4.4
Matriks Defuzzifier

| | | | | | | |
|--------|--------|-----------------|-----------------|---------------|--------------|---------------------|
| Jernih | | Low | LowMed | Normal | HighMed | High |
| | Bening | K. Tohor banyak | K. Tohor sedang | Normal | Tawas sedang | Tawas banyak |
| | Keruh | Buang | Buang | Tawas sedikit | Tawas sedang | Tawas banyak |
| | KeruhH | Buang | Buang | Buang | Tawas banyak | Tawas banyak sekali |

| | | |
|-----------------------|--------|--------|
| DO (dissolved oxigen) | Rendah | Tinggi |
| | Aerasi | Normal |

4.3.5 Pembacaan ADC

Flowchart pembacaan ADC:



Gambar 4.18
Diagram Alir Pembacaan ADC

BAB V

PENGUJIAN DAN PENGUKURAN

Pada bab ini dilakukan pengujian dan pengukuran dari modul-modul sistem yang telah dibuat, sehingga dengan demikian diperoleh keterangan tentang unjuk kerja dari sistem yang dirancang.

5.1 Pengujian Masing-Masing Modul

5.1.1 Modul Paralel Interface (PPI card 8255)

Pada sistem ini digunakan satu card paralel interface 8255. Pengecekan dilakukan dengan menghubungkan port ke display led 8 x 3. Data yang dikirimkan dapat dilihat pada nyala-tidaknya display led.

5.1.2 Modul ADC

Pengujian modul ADC meliputi 2 hal, yaitu pengujian *timing control* dan hasil konversi. Pengujian *timing control* dilakukan dengan menggunakan oscilloscope. Sinyal status konversi data dapat dilihat apakah sudah bekerja dengan baik. Pengujian hasil konversi dibandingkan dengan hasil penunjukkan avometer.

5.1.3 Modul Fuzzy Card NLX230

Pengujian pada card ini meliputi pengujian download, upload, dan running per frame. Untuk mengubah term dan rule fuzzy menjadi data bitmap (biner) digunakan software *development system* (*American Neurologix*). Data ini di-*download* ke internal register NLX230

dan di-*upload/verify* apakah data internal register sudah sama dengan data bitmap semula. Pengujian running per frame dibandingkan dengan hasil *software development system*.

5.1.4 Modul Driver katup

Pengujian modul ini dengan cara menghubungkan langsung ke katup. Kontrol pemilihan katup adalah port A card PPI bit 3 - 7.

5.1.5 Modul Instrumentasi pH

Pengujian modul ini dengan melihat tegangan output modul setelah dicelupkan ke dalam beberapa air dengan pH berlainan. Kemudian dikalibrasi dengan pH meter.

5.1.6 Modul Instrumentasi DO

Pengujian modul ini dengan melihat tegangan output modul setelah dicelupkan ke dalam beberapa air dengan DO yang sudah diketahui berdasarkan metode Winkler. Kemudian dikalibrasi.

5.1.7 Modul Instrumentasi Kejernihan

Pengujian modul ini dengan melihat tegangan output modul setelah dimasukkan ke dalam air yang sudah diketahui tingkat kejernihannya berdasarkan pembacaan turbidity meter. Kemudian dikalibrasi.

5.2 Kalibrasi Modul-Modul Instrumentasi

Untuk memperoleh data yang akurat, maka proses pengukuran dilakukan secara berulang-ulang, dibandingkan dengan hasil pengukuran standar atau hasil pengukuran alat ukur.

5.2.1 Kalibrasi Modul pH

Pada pengukuran pH yaitu pengukuran derajat keasaman dan kebasaan air minum atau baku dipergunakan elektroda gelas yang dalam pengukurannya akan mengalami perubahan tegangan output. Untuk mendapatkan hasil pengukuran yang benar perlu dilakukan koreksi terhadap pembacaan pH meter. Hasil pengukuran disajikan pada tabel di bawah ini.

Tabel 5.1
Hasil pengukuran pH

| pH terukur | Pembacaan ADC (V) | | | | | | | | | |
|---------------|-------------------|------|------|------|------|------|------|------|------|------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 10,6 | 4,16 | 4,16 | 4,16 | 4,16 | 4,15 | 4,16 | 4,15 | 4,15 | 4,16 | 4,16 |
| 9,6 | 3,53 | 3,53 | 3,53 | 3,51 | 3,53 | 3,51 | 3,55 | 3,55 | 3,53 | 3,53 |
| 4,7 | 0,67 | 0,67 | 0,69 | 0,69 | 0,69 | 0,65 | 0,67 | 0,67 | 0,67 | 0,69 |
| 4 | 0,35 | 0,35 | 0,35 | 0,37 | 0,37 | 0,37 | 0,37 | 0,37 | 0,37 | 0,37 |

Dari tabel di atas dapat dibuat grafik dengan pendekatan secara regresi linier, dimana persamaan garis lurus regresi dari hasil pengukuran dapat dicari dengan rumus :

$$Y = mX + b$$

dimana:

$$m = \frac{n\sum xy - \sum x \sum y}{n\sum x^2 - (\sum x)^2}$$

$$b = \frac{\sum y \sum x^2 - \sum xy \sum x}{n\sum x^2 - (\sum x)^2}$$

dengan

n = banyaknya data

X = output ADC

Y = pH yang terukur

Jadi persamaan garis linier hasil regresi kalibrasi pengukuran pH adalah :

$$Y = 32,72 X - 112,36$$

5.2.2 Kalibrasi Modul Kejernihan

Kalibrasi kejernihan dilakukan terhadap kejernihan air yang sudah diukur dengan turbidity meter, yaitu alat untuk mengukur tingkat kejernihan air. Hasil pengukuran disajikan pada tabel 5.2 di bawah.

Tabel 5.2
Hasil pengukuran kejernihan

| Turbidity meter | Pembacaan ADC (V) | | | | | | | | | |
|--------------------|-------------------|------|------|------|------|------|------|------|------|------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 0,92 NTU | 3,65 | 3,67 | 3,62 | 3,65 | 3,64 | 3,63 | 3,65 | 3,66 | 3,65 | 3,67 |
| 1,5 NTU | 3,54 | 3,57 | 3,54 | 3,55 | 3,57 | 3,55 | 3,56 | 3,56 | 3,54 | 3,54 |
| 14,4 NTU | 1,20 | 1,23 | 1,25 | 1,24 | 1,25 | 1,25 | 1,24 | 1,23 | 1,25 | 1,24 |

Dari tabel di atas dapat dibuat grafik dengan pendekatan secara regresi linier, dimana garis lurus regresi hasil pengukuran di atas dapat dicari dengan rumus:

$$Y = mX + b$$

dimana:

$$m = \frac{n \sum xy - \sum x \sum y}{n \sum x^2 - (\sum x)^2}$$

$$b = \frac{\sum y \sum x^2 - \sum xy \sum x}{n \sum x^2 - (\sum x)^2}$$

dengan

n = banyaknya data

X = kejernihan air (NTU)

Y = kejernihan terukur

Jadi persamaan garis linier dari hasil regresi kalibrasi pengukuran adalah:

$Y = (-0,18)X + 3,82$

5.2.3 Kalibrasi Modul DO

Kalibrasi modul kadar oksigen terlarut (DO) ini dilakukan terhadap kadar oksigen air yang sudah diketahui dengan menggunakan metode Winkler. Hasil pengukuran disajikan pada tabel di bawah ini.

Tabel 5.3
Hasil pengukuran DO

| Metoda Winkler (mg O ₂ /l) | Pembacaan ADC (V) | | | | | | | | | |
|---------------------------------------|-------------------|------|------|------|------|------|------|------|------|------|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 0 | 2,85 | 2,74 | 2,8 | 2,87 | 2,89 | 2,83 | 2,83 | 2,85 | 2,81 | 2,8 |
| 2,62 | 3,17 | 3,1 | 3,2 | 3,22 | 3,13 | 3,14 | 3,15 | 3,15 | 3,17 | 3,18 |
| 5,25 | 3,43 | 3,40 | 3,48 | 3,51 | 3,44 | 3,48 | 3,45 | 3,48 | 3,47 | 3,49 |

Dari tabel di atas dapat dibuat grafik dengan pendekatan secara regresi linier, dimana garis lurus regresi hasil pengukuran di atas dapat dicari dengan rumus:

$Y = mX + b$

dimana:

$$m = \frac{n\sum xy - \sum x \sum y}{n\sum x^2 - (\sum x)^2}$$

$$b = \frac{\sum y \sum x^2 - \sum xy \sum x}{n\sum x^2 - (\sum x)^2}$$

dengan

n = banyaknya data

X = kejernihan air (NTU)

Y = kejernihan terukur

Jadi persamaan garis linier dari hasil regresi kalibrasi pengukuran adalah:

$Y = 8,1 X - 22,9$

5.3 Data Percobaan Penetralan

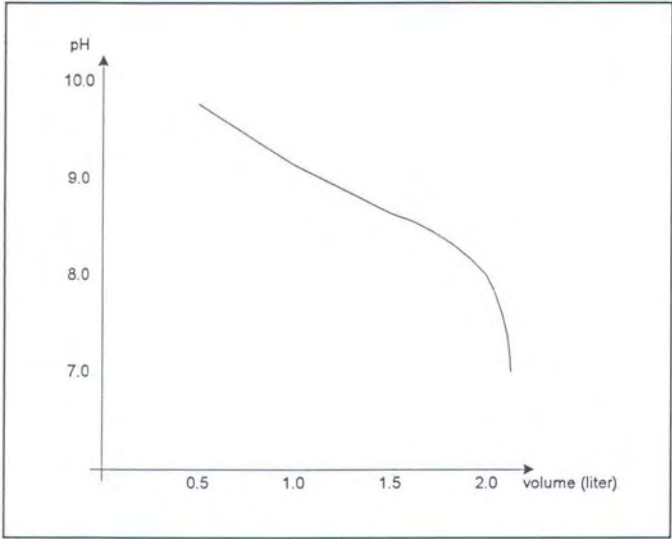
5.3.1 Penetralan Dari Keadaan Basa

Air bersifat basa dengan volume 7 liter dan pH awal 10 dinetralkan dengan air bersifat asam (tawas terlarut) dengan pH 4,1.

Tabel 5.4

Penetralan Dari Keadaan Basa

| Penambah (ml) | pH air |
|---------------|--------|
| 500 | 9,9 |
| 500 | 9,9 |
| 500 | 9,8 |
| 500 | 9,8 |
| 500 | 9,9 |
| 1000 | 9,6 |
| 1000 | 9,4 |
| 1000 | 9,5 |
| 1000 | 9,4 |
| 1000 | 9,4 |
| 1500 | 9 |
| 1500 | 9,1 |
| 1500 | 9,1 |
| 1500 | 9,2 |
| 1500 | 9,1 |
| 2000 | 8,7 |
| 2000 | 8,5 |
| 2000 | 8,6 |
| 2000 | 8,3 |
| 2000 | 8,8 |
| 2500 | 7,3 |
| 2500 | 7,4 |
| 2500 | 7,6 |
| 2500 | 6,8 |
| 2500 | 6,9 |



Gambar 5.1
Grafik Penetralan dari Keadaan Basa

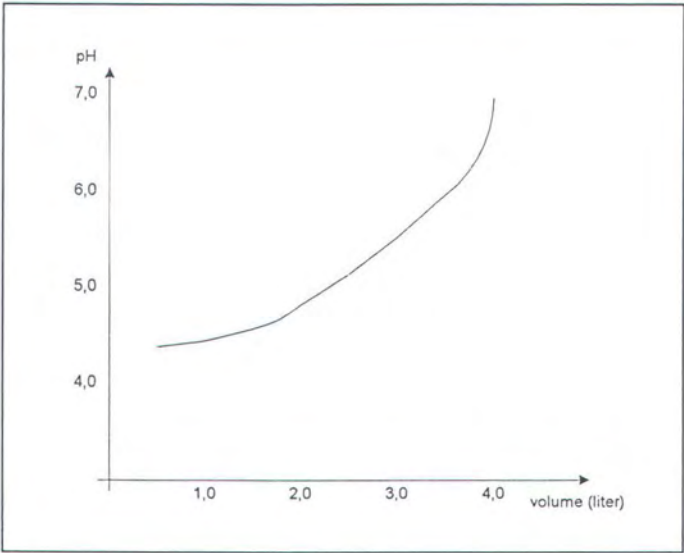
5.3.2 Penetralan Dari Keadaan Asam

Air bersifat asam dengan volume 7 liter dan pH awal 4,4 dinetralkan dengan air bersifat basa (kapur tohor terlarut) dengan pH 9,9.

Tabel 5.5
Penetralan Dari Keadaan Asam

| Penambah (ml) | pH air |
|---------------|--------|
| 500 | 4,4 |
| 500 | 4,4 |
| 500 | 4,4 |
| 500 | 4,4 |
| 500 | 4,4 |
| 1000 | 4,4 |
| 1000 | 4,5 |
| 1000 | 4,4 |
| 1000 | 4,4 |
| 1000 | 4,5 |
| 1500 | 4,5 |
| 1500 | 4,5 |
| 1500 | 4,6 |
| 1500 | 4,4 |
| 1500 | 4,5 |
| 2000 | 4,7 |

| | |
|------|-----|
| 2000 | 4,7 |
| 2000 | 4,8 |
| 2000 | 4,9 |
| 2000 | 4,8 |
| 2500 | 5 |
| 2500 | 5,1 |
| 2500 | 5,3 |
| 2500 | 5,3 |
| 2500 | 5,2 |
| 3000 | 5,4 |
| 3000 | 5,5 |
| 3000 | 5,6 |
| 3000 | 5,5 |
| 3000 | 5,5 |
| 3500 | 5,8 |
| 3500 | 5,9 |
| 3500 | 6 |
| 3500 | 5,9 |
| 3500 | 6,1 |
| 4000 | 6,9 |
| 4000 | 7 |
| 4000 | 6,8 |
| 4000 | 6,7 |
| 4000 | 6,9 |



Gambar 5.2
Grafik Penetralan dari Keadaan Asam

5.4 Data Percobaan Penambahan Larutan Tawas atau Kapur Tohor

Penambahan larutan tawas dan kapur tohor yang digunakan sebagai pengkondisi air pada alat ini diatur dengan mengaktifkan katup-katup. Pengambilan data volume penambahan dilakukan dengan keadaan awal bak penampung penuh dan dengan mengatur durasi waktu buka katup.

Tabel 5.6
Data Volume Penambahan

| Durasi buka katup (detik) | Volume (ml) | Durasi buka katup (detik) | Volume (ml) |
|---------------------------|-------------|---------------------------|-------------|
| 15 | 550 | 60 | 2200 |
| 15 | 575 | 60 | 2100 |
| 15 | 550 | 60 | 2150 |
| 15 | 560 | 60 | 2050 |
| 15 | 600 | 60 | 2100 |
| 30 | 1250 | 120 | 4050 |
| 30 | 1200 | 120 | 3980 |
| 30 | 1275 | 120 | 4000 |
| 30 | 1200 | 120 | 4100 |
| 30 | 1150 | 120 | 4100 |

BAB VI

PENUTUP

6.1 Kesimpulan

Dari hasil studi analisa secara elektronik dan kimiawi serta pembuatan alat yang dirancang dapat ditarik beberapa kesimpulan sebagai berikut:

1. Sistem kontrol dengan menggunakan logika fuzzy mempunyai beberapa keunggulan sebagai berikut:
 - ◆ Tidak memerlukan model matematis sehingga mempermudah proses perancangan.
 - ◆ Aturan fuzzy bersifat intuitif sehingga mudah dimengerti.
 - ◆ Fleksibilitas tinggi karena mudah untuk mengubah aturan/rule sesuai dengan keperluan.
2. Penggunaan chip fuzzy NLX230 mempunyai beberapa keterbatasan di antaranya sebagai berikut:
 - ◆ Pengoperasiannya pada sistem ini tidak dapat secara per frame, tetapi secara per channel dan ini menyebabkan mode output yang digunakan hanya *mode immediate*.
 - ◆ Bentuk fungsi membership hanya dua macam, yaitu *tipe exclusive* dan *inclusive*.
 - ◆ Metode defuzzifier tidak dapat diubah.
3. Keandalan transduser yang digunakan merupakan faktor yang sangat utama untuk menentukan unjuk kerja dari sistem yang dibuat.
4. Pengujian kualitas air - pH, DO, dan kejernihan - dapat dilakukan secara bersamaan.

5. Dalam air terdapat unsur-unsur carbonat (CO_3), bicarbonat (HCO_3), hidroxy, dan ion OH^- yang berpengaruh pada proses penetralan.
6. Ketelitian kapasitansi meter untuk mendeteksi jumlah volume yang ditambahkan dipengaruhi ke-homogen-an larutan sebagai dielektrik dan interferensi keadaan sekitar.

6.2 Saran

Untuk pengembangan lebih lanjut kami sampaikan beberapa saran sebagai berikut:

1. Sistem rancangan kontrol kualitas air ini merupakan ide ilmiah, sehingga untuk penerapan di lapangan masih diperlukan pengkajian lebih mendalam.
2. Untuk mendapatkan kualitas air yang baik dibutuhkan transduser dengan tingkat akurasi yang handal.

“ Jika Anda tidak ingin dilupakan orang segera setelah meninggal dunia, maka tulislah sesuatu yang patut dibaca atau berbuatlah sesuatu yang patut diabadikan dalam tulisan “

- *Benyamin Franklin* -



DAFTAR PUSTAKA

DAFTAR PUSTAKA

- ◆ Coughlin, Robert F., dan Driscoll, Frederick F., PENGUAT DAN RANGKAIAN TERPADU LINEAR, edisi kedua, Penerbit Erlangga, 1992.
- ◆ Day Jr, R.A., Underwood, A.L., (Drs. Soendoro, Ny. Widaningsih W., B.A., Dra. Sri Rahadjeng S. ; penerjemah), ANALISA KIMIA KUANTITATIF, edisi keempat, Penerbit Erlangga, 1981.
- ◆ Fair, Gordon M., Geyer, John C., Okun, Daniel A., WATER AND WASTEWATER ENGINEERING, John Wiley and Sons, Inc., 1968.
- ◆ Gayakwad, Ramakant A., OP-AMPS & LINEAR INTEGRATED CIRCUIT, second edition, Prentice Hall of India, New Delhi, 1989.
- ◆ Hall, Dauglas V., MICROPROCESSOR AND INTERFACING, McGraw-Hill Book Co., 1986.
- ◆ HM, Jogyianto., TEORI DAN APLIKASI PROGRAM KOMPUTER BAHASA PASCAL, jilid 1 dan 2, Andi Offset, Yogyakarta, 1991.
- ◆ Hughes, Fredrick W., PANDUAN OP-AMP, P.T. Elex Media Komputindo, Kelompok Gramedia-Jakarta, 1990.
- ◆ Keenan, Charles W., Kleinfelter, Donald C., Wood, Jesse H., (Aloysius Hadyana Pudjaatmaka, penerjemah), KIMIA UNTUK UNIVERSITAS, edisi ke-6, Penerbit Erlangga, 1989.
- ◆ Klir, George J., dan Folger, Tina A., FUZZY SETS, UNCERTAINTY, AND INFORMATION, Simon & Schuster (Asia) Pte Ltd, 1992.
- ◆ Maloney, Timothy J., INDUSTRIAL SOLID-STATE ELECTRONICS DEVICES AND SYSTEMS, second edition, Prentice Hall, 1986.

- ◆ Pescok, Robert L., MODERN METHOD OF CHEMICAL ANALYSIS, second edition, John Wiley & Sons Inc., 1986.
- ◆ Sawyer, Clair N., CHEMISTRY FOR SANITARY ENGINEERS, second edition, McGraw-Hill Book Company, Kogakusha Company Ltd., 1987.
- ◆ S, Wasito, VADEMEKUM ELEKTRONIKA, P.T. Gramedia, Jakarta, 1984.
- ◆ Steeman, J.P.M., DATA SHEET BOOK 2, Elex Media Komputindo, Jakarta, 1989.
- ◆ Yan, Jun, Ryan, Michael, USING FUZZY LOGIC, Prentice Hall, 1994
- ◆, CMOS DATABOOK, National Semiconductor Corporation, 1984.
- ◆, FUZZY MICROCONTROLLER DEVELOPMENT SYSTEM, American Neurologix Inc, 1992.
- ◆, LINEAR 1, National Semiconductor, California, 1982.
- ◆, LINEAR 2, National Semiconductor, California, 1988.
- ◆, NEUROLOGIX DATA SHEET NLX230, American Neurologix Inc, 1992.
- ◆, SOFTWARE TUTORIAL FUZZY, Motorola.

RIWAYAT HIDUP



YUSUF LESTANTO dilahirkan di Surabaya pada tanggal 2 Mei 1971, putra kedua dari empat bersaudara dari Bapak Zuhdi Anwar dan Ibu Indijah, yang bertempat tinggal di jalan Banyu Urip Lor III A/8 Surabaya.

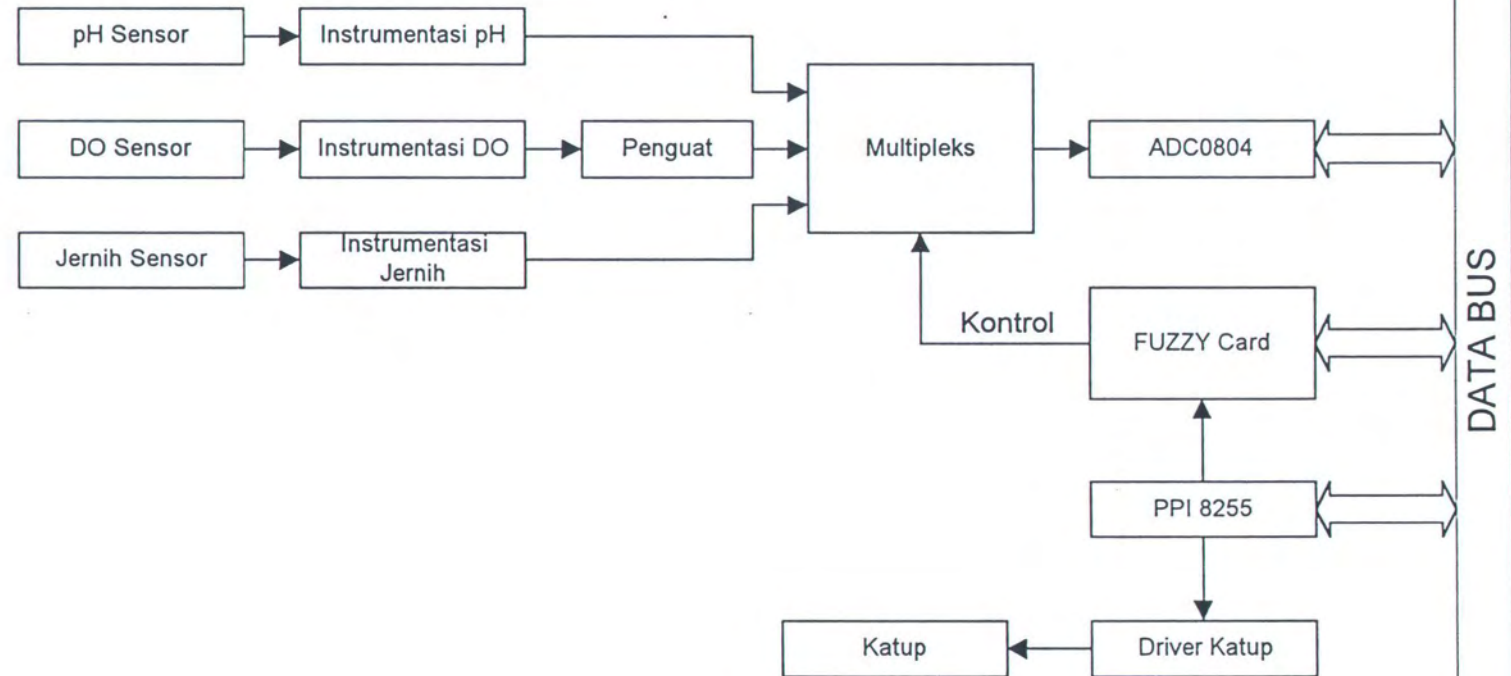
Terdaftar sebagai mahasiswa Institut Teknologi Sepuluh Nopember Surabaya, Jurusan Teknik Elektro pada tahun 1990 dengan NRP. 2902201468.

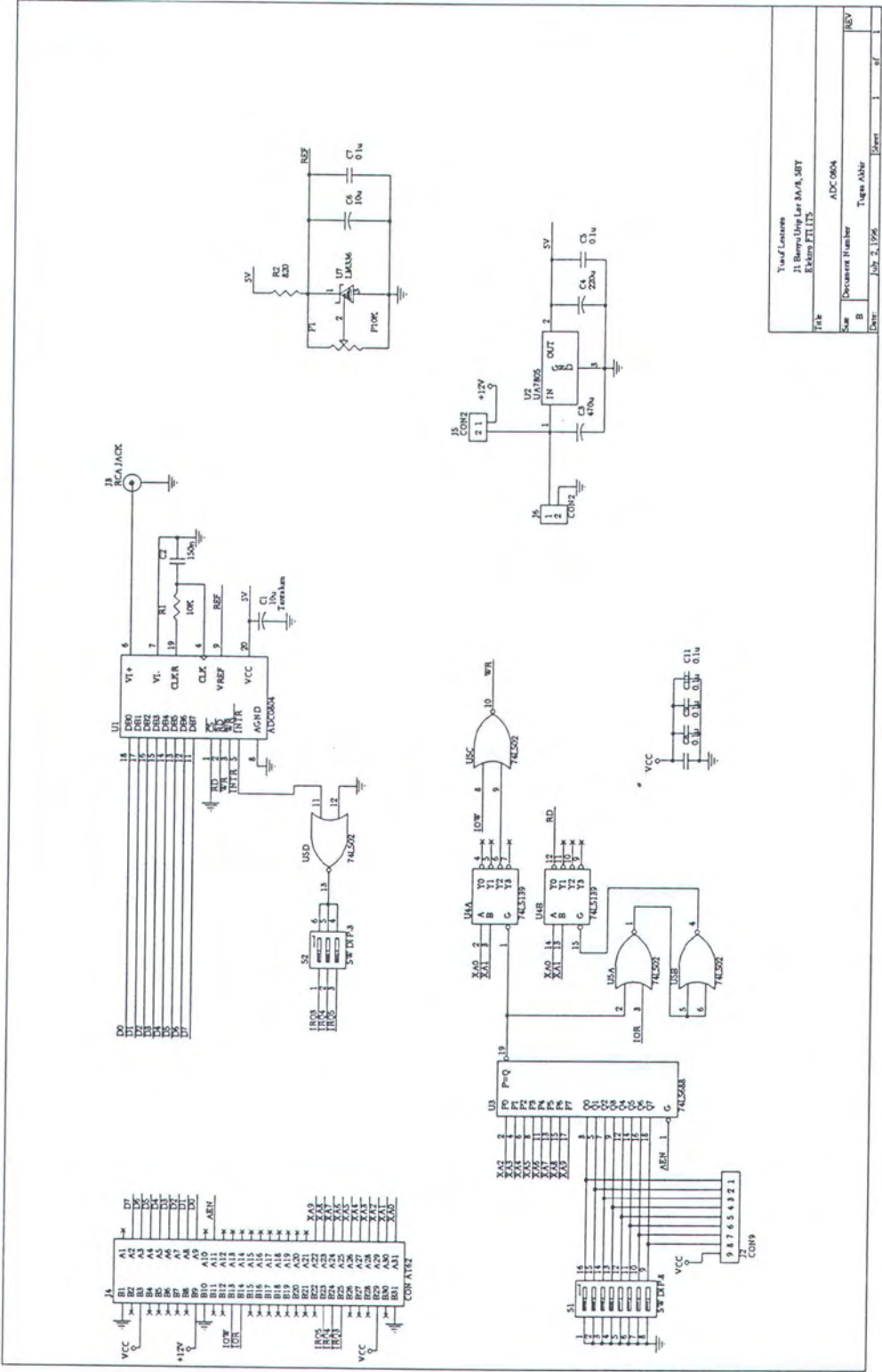
Pendidikan yang pernah ditempuh :

- ♦ TK Putra Bangsa, tahun 1977 - 1978
- ♦ SDN Sawahan VIII, tahun 1978 - 1984
- ♦ SMPN 3 Surabaya, tahun 1984 - 1987
- ♦ SMAN 5 Surabaya, tahun 1987 - 1990
- ♦ Jurusan Teknik Elektro ITS pada tahun 1990 dan diharapkan lulus pada periode Nopember 1996.

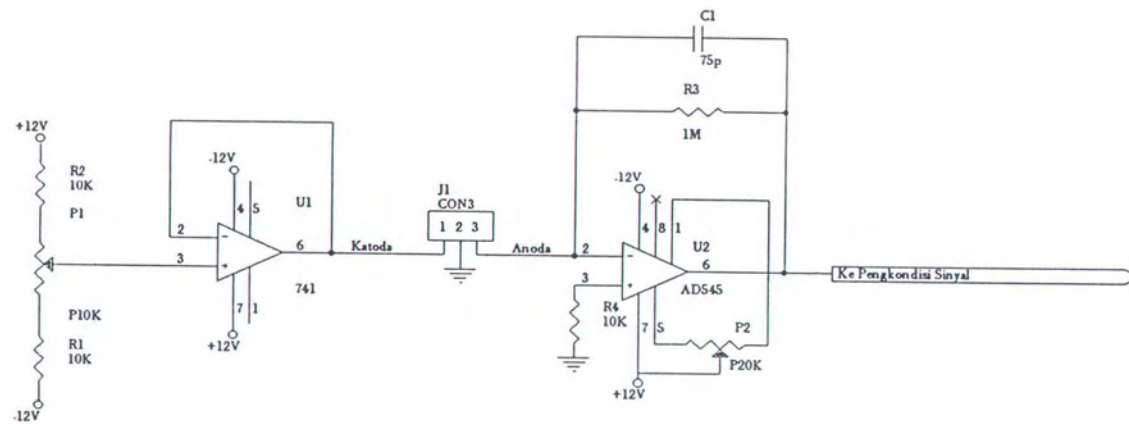
Selama menjadi mahasiswa pernah aktif :

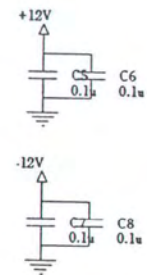
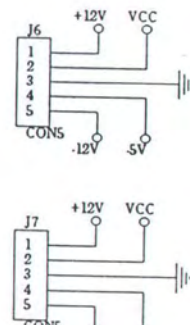
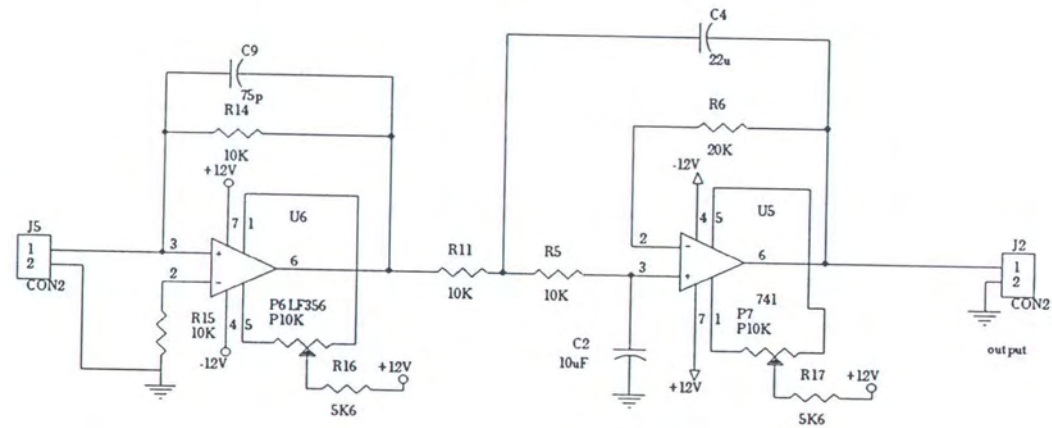
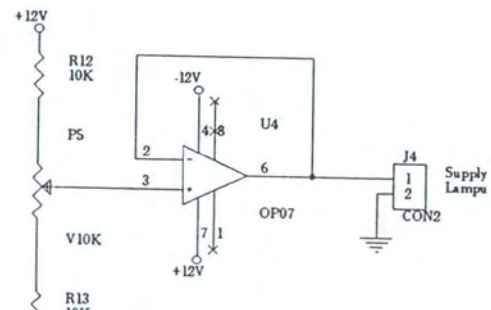
- ♦ Koordinator Kursus Interfacing to IBM PC dan Bahasa Assembly, tahun 1994 - 1995.
- ♦ Koordinator Praktikum Rangkaian Listrik Mahasiswa ITPS, tahun 1994.
- ♦ Asisten Praktikum di lingkungan bidang studi Elektronika.

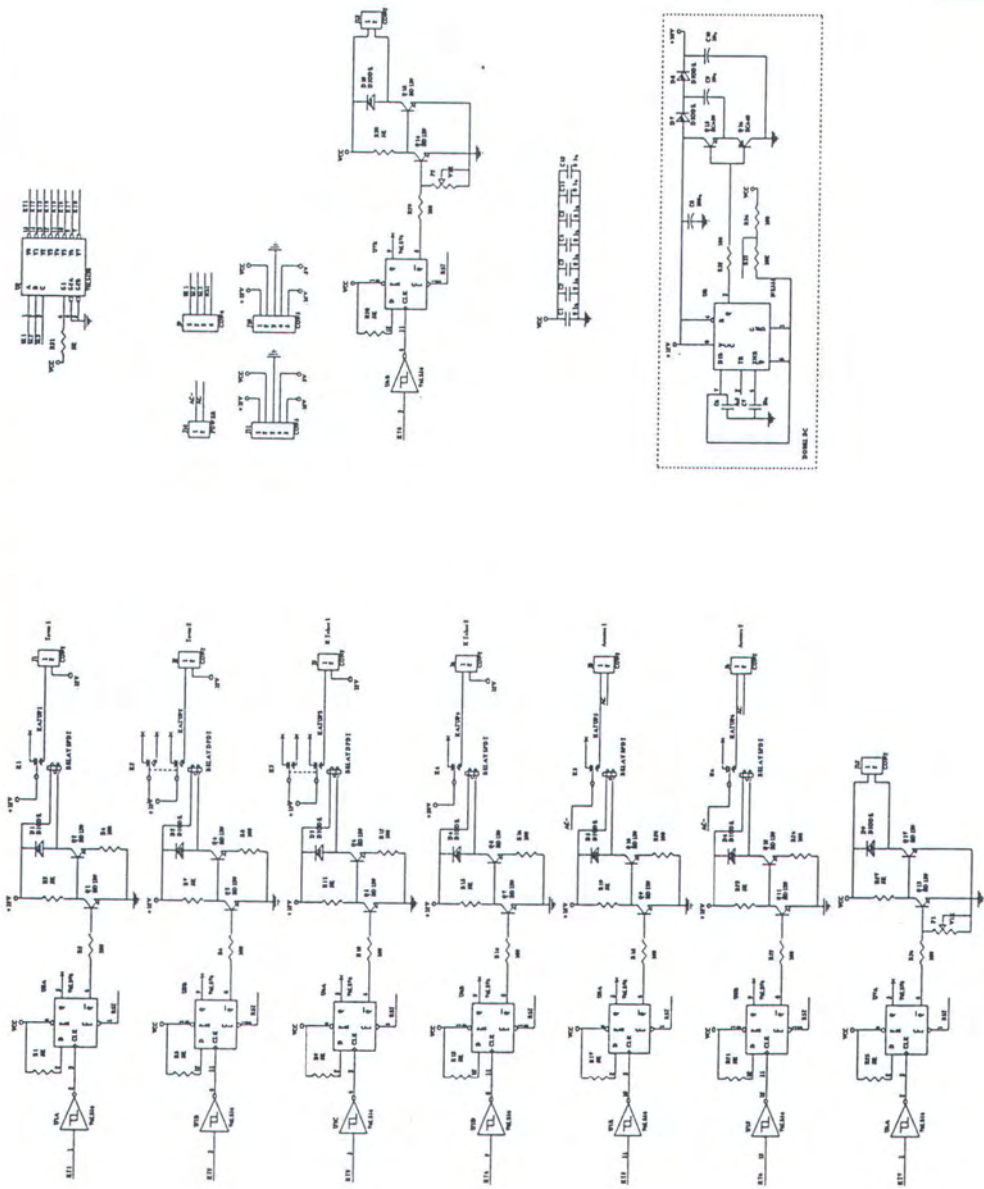




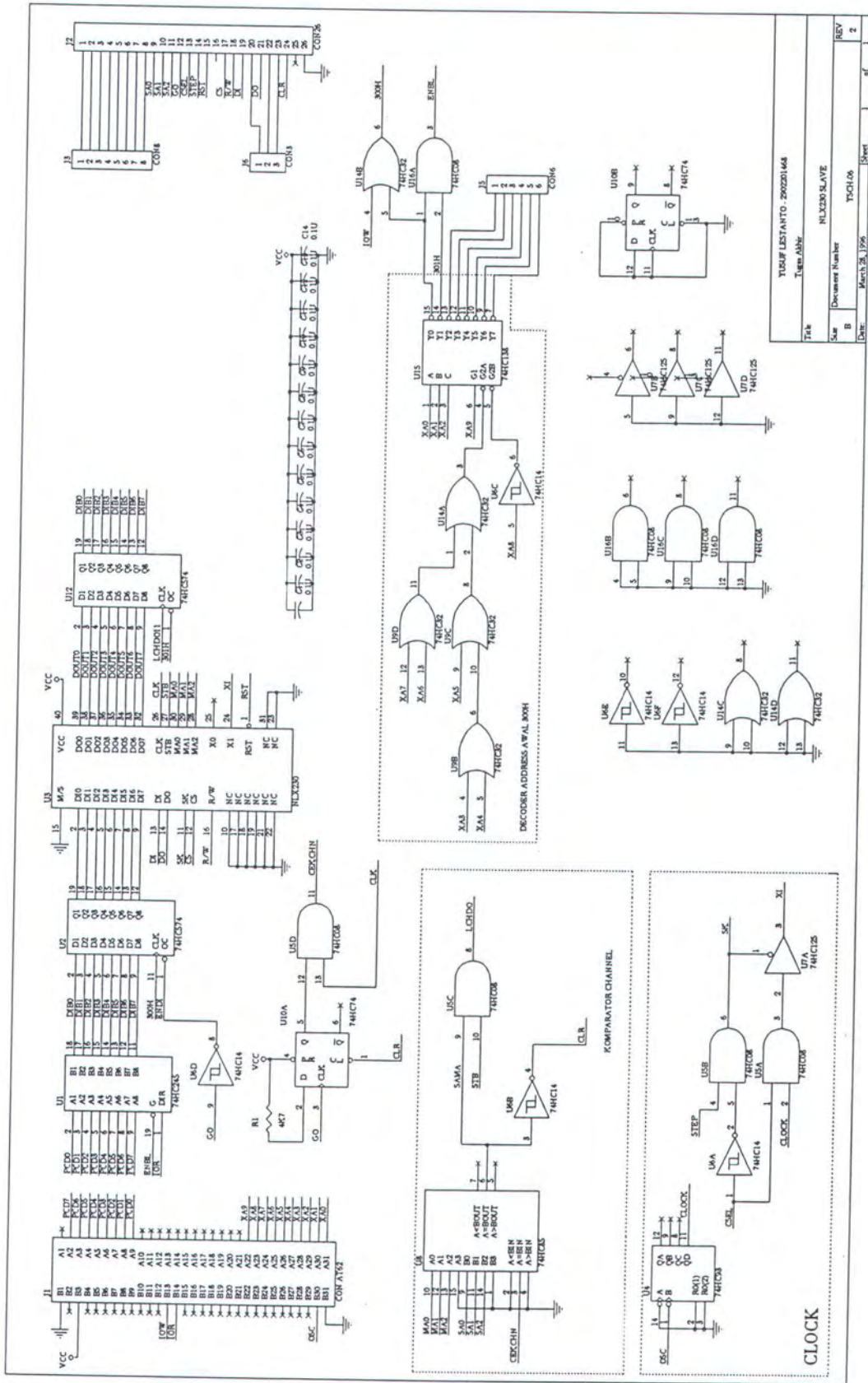
| | | | |
|-------------------------------|--|--|--|
| Title of Document | | | |
| 11. Binary Logic for M.A. SET | | | |
| Etkin F11.115 | | | |
| File | | | |
| See Document Number | | | |
| ADC 0804 | | | |
| B | | | |
| Tuzun Akbir | | | |
| Date | | | |
| July 2, 1998 | | | |
| Sheet | | | |
| 1 of | | | |



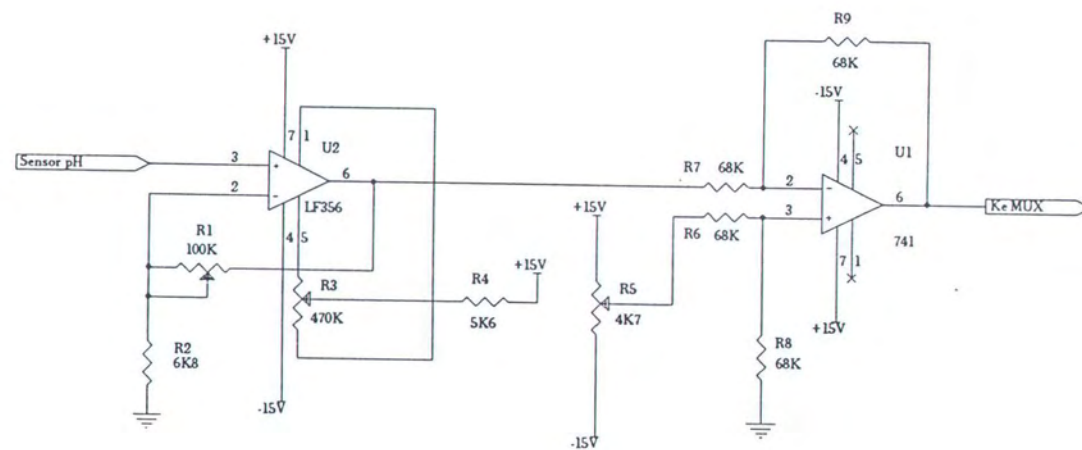


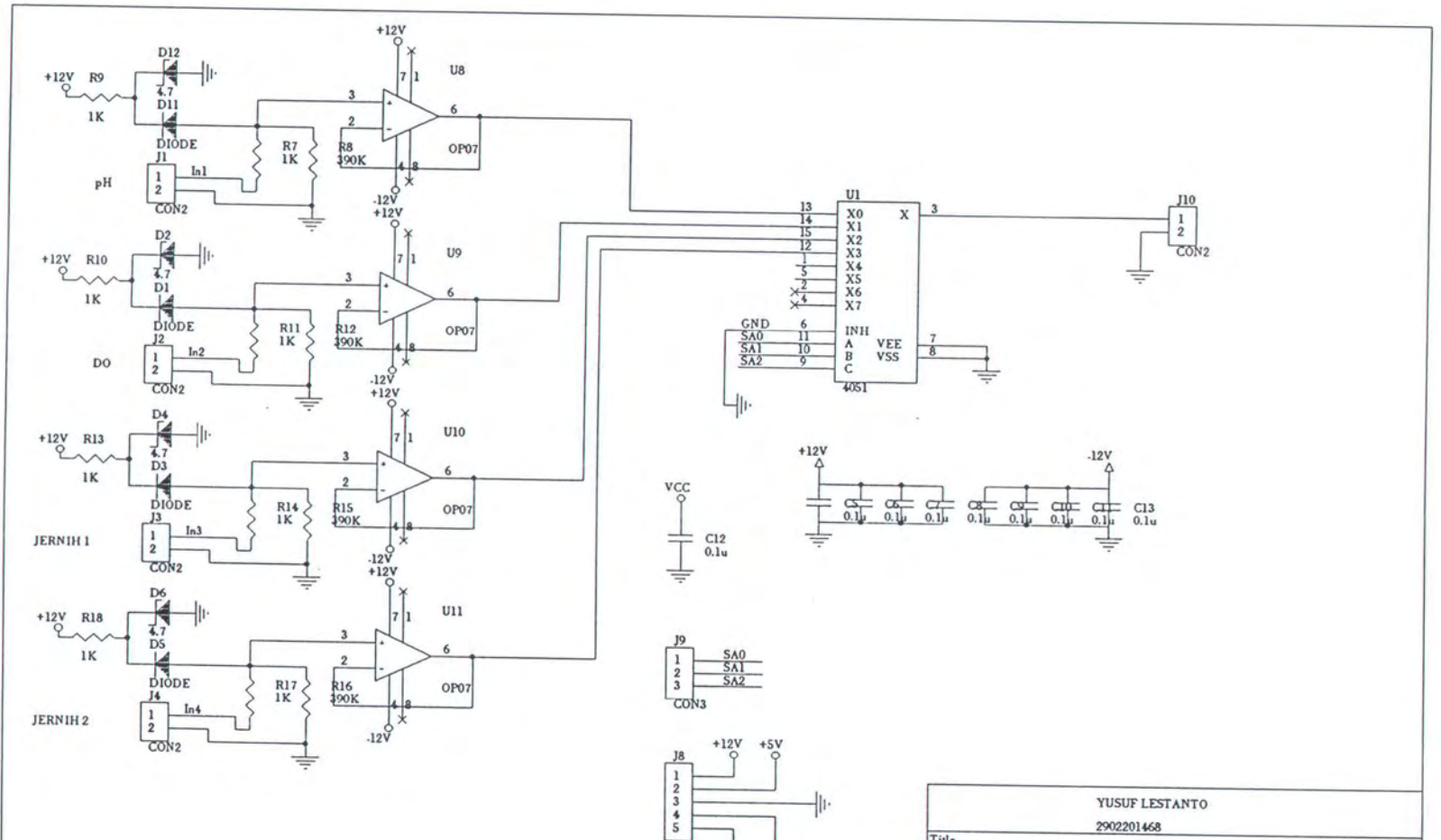


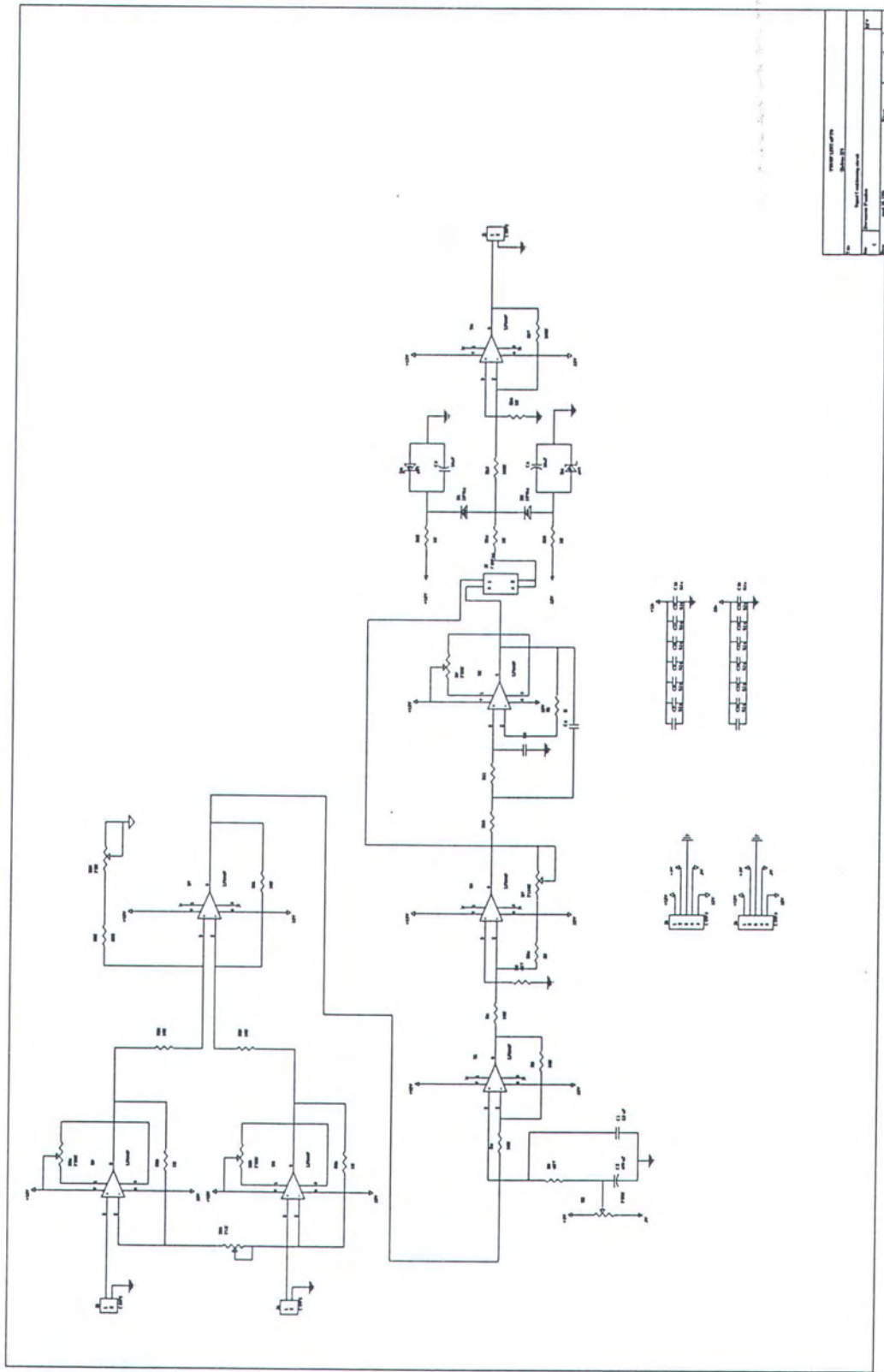
| | |
|--|--|
| Nome: LUIS ALBERTO Matrícula: 10010000000000000000 Data: 10/01/2020 | |
| Assinatura: | |
| Assinatura do orientador: | |
| Assinatura do avaliador: | |
| Assinatura do professor: | |



| | | |
|------------|----------------|---------------|
| Title | | 74HC230 SLAVE |
| Doc Number | 74HC230 | |
| Rev | 2 | |
| Date | March 28, 1996 | |
| Sheet | of | |







| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|-----|
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | 65 | 66 | 67 | 68 | 69 | 70 | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 | 84 | 85 | 86 | 87 | 88 | 89 | 90 | 91 | 92 | 93 | 94 | 95 | 96 | 97 | 98 | 99 | 100 |
| Rev | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 60 | 61 | 62 | 63 | 64 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

File Bitmap Term dan Rule :

| | | | | | |
|--------|--------|--------|--------|--------|--------|
| 0/00; | 2f/3e; | 5e/00; | 8d/00; | bc/00; | eb/00; |
| 1/00; | 30/1e; | 5f/00; | 8e/00; | bd/00; | ec/00; |
| 2/00; | 31/5e; | 60/00; | 8f/00; | be/00; | ed/00; |
| 3/00; | 32/0a; | 61/00; | 90/00; | bf/00; | ee/00; |
| 4/00; | 33/5e; | 62/00; | 91/00; | c0/00; | ef/00; |
| 5/00; | 34/00; | 63/00; | 92/00; | c1/00; | f0/00; |
| 6/00; | 35/00; | 64/00; | 93/00; | c2/00; | f1/00; |
| 7/00; | 36/00; | 65/00; | 94/00; | c3/00; | f2/00; |
| 8/00; | 37/00; | 66/00; | 95/00; | c4/00; | f3/00; |
| 9/00; | 38/00; | 67/00; | 96/00; | c5/00; | f4/00; |
| a/00; | 39/00; | 68/00; | 97/00; | c6/00; | f5/00; |
| b/00; | 3a/00; | 69/00; | 98/00; | c7/00; | f6/00; |
| c/f8; | 3b/00; | 6a/00; | 99/00; | c8/00; | f7/00; |
| d/ff; | 3c/00; | 6b/00; | 9a/00; | c9/01; | f8/00; |
| e/80; | 3d/00; | 6c/00; | 9b/00; | ca/02; | f9/00; |
| f/02; | 3e/00; | 6d/00; | 9c/00; | cb/00; | fa/00; |
| 10/32; | 3f/00; | 6e/00; | 9d/00; | cc/00; | fb/00; |
| 11/32; | 40/00; | 6f/00; | 9e/00; | cd/00; | fc/00; |
| 12/32; | 41/ce; | 70/00; | 9f/00; | ce/00; | fd/00; |
| 13/00; | 42/e7; | 71/00; | a0/00; | cf/00; | fe/00; |
| 14/00; | 43/00; | 72/00; | a1/00; | d0/00; | ff/00; |
| 15/00; | 44/19; | 73/00; | a2/00; | d1/00; | |
| 16/00; | 45/32; | 74/00; | a3/00; | d2/00; | |
| 17/00; | 46/46; | 75/00; | a4/00; | d3/00; | |
| 18/05; | 47/00; | 76/00; | a5/00; | d4/00; | |
| 19/08; | 48/d8; | 77/00; | a6/00; | d5/00; | |
| 1a/0a; | 49/ce; | 78/00; | a7/00; | d6/00; | |
| 1b/00; | 4a/00; | 79/00; | a8/00; | d7/00; | |
| 1c/00; | 4b/00; | 7a/00; | a9/00; | d8/00; | |
| 1d/00; | 4c/00; | 7b/00; | aa/00; | d9/00; | |
| 1e/00; | 4d/00; | 7c/00; | ab/00; | da/00; | |
| 1f/00; | 4e/00; | 7d/00; | ac/00; | db/00; | |
| 20/1e; | 4f/00; | 7e/00; | ad/00; | dc/00; | |
| 21/1e; | 50/00; | 7f/00; | ae/00; | dd/00; | |
| 22/3c; | 51/00; | 80/00; | af/00; | de/00; | |
| 23/1e; | 52/00; | 81/01; | b0/00; | df/00; | |
| 24/5a; | 53/00; | 82/02; | b1/00; | e0/00; | |
| 25/1e; | 54/00; | 83/04; | b2/00; | e1/00; | |
| 26/78; | 55/00; | 84/08; | b3/00; | e2/00; | |
| 27/1e; | 56/00; | 85/10; | b4/00; | e3/00; | |
| 28/96; | 57/00; | 86/20; | b5/00; | e4/00; | |
| 29/1e; | 58/00; | 87/40; | b6/00; | e5/00; | |
| 2a/00; | 59/00; | 88/80; | b7/00; | e6/00; | |
| 2b/3e; | 5a/00; | 89/00; | b8/00; | e7/00; | |
| 2c/28; | 5b/00; | 8a/00; | b9/00; | e8/00; | |
| 2d/3e; | 5c/00; | 8b/00; | ba/00; | e9/00; | |
| 2e/1e; | 5d/00; | 8c/00; | bb/00; | ea/00; | |

Uses Graph,Crt,dos,mouseku:

```
A :integer = $308;  
B :integer = $309;  
C :integer = $30A;  
CW:integer = $30B;  
TombolOn  = true;  
TombolOff = false;  
Aktif     = true;
```

```
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$f8,$ff,$80,$02 ;
{ 0 1 2 3 4 5 6 7 8 9 A B C D E F }
$32,$32,$32,$00,$00,$00,$00,$00,$00,$05,$08,$0a,$00,$00,$00,$00,$00,$00,
{10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F}
$1e,$1e,$3c,$1e,$5a,$1e,$78,$1e,$96,$1e,$00,$3e,$28,$3e,$1e,$3f,
{20 21 22 23 24 25 26 27 28 29 2A 2B 2C 2D 2E 2F}
$1e,$5e,$0a,$5e,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{30 31 32 33 34 35 36 37 38 39 3A 3B 3C 3D 3E 3F}
$00,$ce,$e7,$00,$19,$32,$46,$00,$d8,$ce,$00,$00,$00,$00,$00,$00,$00,$00,
{40 41 42 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{50 51 52 53 54 55 56 57 58 59 5A 5B 5C 5D 5E 5F}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{60 61 62 63 64 65 66 67 68 69 6A 6B 6C 6D 6E 6F}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{70 71 72 73 74 75 76 77 78 79 7A 7B 7C 7D 7E 7F}
$00,$01,$02,$04,$08,$10,$20,$40,$80,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{80 81 82 83 84 85 86 87 88 89 8A 8B 8C 8D 8E 8F}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{90 91 92 93 94 95 96 97 98 99 9A 9B 9C 9D 9E 9F}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{A0 A1 A2 A3 A4 A5 A6 A7 A8 A9 AA AB AC AD AE AF}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{B0 B1 B2 B3 B4 B5 B6 B7 B8 B9 BA BB BC BD BE BF}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$01,$02,$00,$00,$00,$00,$00,$00,$00,
{C0 C1 C2 C3 C4 C5 C6 C7 C8 C9 CA CB CC CD CE CF}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{D0 D1 D2 D3 D4 D5 D6 D7 D8 D9 DA DB DC DD DE DF}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{E0 E1 E2 E3 E4 E5 E6 E7 E8 E9 EA EB EC ED EE EF}
$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,$00,
{F0 F1 F2 F3 F4 F5 F6 F7 F8 F9 FA FB FC FD FE FF}
```

Minum');
Start : integer = \$30E;
Baca : integer = \$30C;

```
AtrTombol = record
    Absis, Ordinat,
    Lebar, Tinggi : word;
    Tebal          : byte;
end;
PushButton = array[1..6] of AtrTombol;
```

```

Pelak           : PushButton;
temp, Size,down,ambil,simpan,Go,J,itung,clear,
dat1,dat2,dat3, oput,,bubar, Kode,Mode,l,selesai,
VolBaku,VolMinum,hit1      : Integer;
BoxDipilih       : byte;
Tombol           : char;
X1,X2,X3         : STRING;
Y,X              : word;
DataADC,NTU,AddTawas,AddTawas1,AddTohor,AddTohor1 : real;

```

```

procedure tawas05;           forward;
procedure DOTawas05;        forward;
procedure tawas15;          forward;
procedure DOTawas15;        forward;
procedure Aerator;          forward;
procedure proses (var AddTawas,AddTohor:real); forward;
procedure proses1 (Var AddTawas1,AddTohor1:real); forward;
procedure Defuzz1 (FuzOutpH,FuzOutDOx,FuzOutJernih:integer;

```

```

var TbhTawas1,TbhTohor1:real);          forward;
procedure Defuzz (FuzOutpH,FuzOutDOx,FuzOutJernih:integer;
var TbhTawas,TbhTohor:real);          forward;
Procedure Hapus(warnalatar,x1,y1,x2,y2:integer);          forward;
Function Kotak(x1,y1,x2,y2,rupa1,rupa2:integer):Integer; forward;
Procedure BuatTombol(warna,X,Y,L,T: word;
Tbl,cat,cat1,font:byte; pesan:string);forward;
Procedure Simulasi (warna,X,Y,L,T : word;
Tbl,cat,cat1,font: byte; pesan:string);forward;
Procedure running(kanal : integer; var otput : real);forward;

```

```

procedure Download;
label gagal,selesai,done,metu;
begin
    BuatTombol(0,530,370,80,35,3,0,15,5,'WAIT' );
    port[CW] := $88;           {PA,PB,PCup = output; PClow = input}
    port[C] := $0;             {clear FMC CS,R/W}
    dat2 := $40;
    port[B] := dat2;           {clear FMC SK}
    port[B] := dat2 or $20;    {raise SK }
    port[C] := 1;              {set FMC CS}
    port[B] := $60;           {set SK}
    for Size:=1 to 256 do      {looping 2048 times}
    begin
        temp := data[Size];    {Get data per 8 bits}
        dat2 := $40;
        dat3 := 1;
    end
end

```



```

for I:=1 to 8 do
begin
    port[B]:=$40;      {clear SK}
    dat3:=dat3 and $1; {set CS}
    down:=temp and $80; {ambil MSB bit}
    down:=down shr 5;   {geser ke kanan mjd bit ke-3}
    dat3:=dat3 or down; {mengubah bit ke-3 dari dat3}
    down:= down shr 2;
    port[B]:=dat2;      {clear SK}
    port[C]:=dat3;      {bit dimasukkan}
    port[B]:=dat2 or $20; {set SK}
    temp:= temp shl 1;
end;
end;
port[B]:=$40;      {clear CLK sel, SK}
port[C]:=0;        {clear FMC CS}

port[B]:=$60;      {set SK}
port[B]:=$40;      {clear SK}
port[C]:=$60;      {set SK}

{ upload }
port[C]:= $03;      {set FMC CS, R/W}
port[B]:= $40;      {clear SK, CLK sel}
dat2:= $40;
for I:=1 to 2 do
begin
    port[B]:= dat2 or $20; {raise SK}
    port[B]:= dat2;
end;
dat2:= dat2 or $20; {set SK}
port[B]:= dat2;
for I:= 1 to 256 do
begin
    ambil := 0;
    simpan:= 0;
    for J:=1 to 8 do
    begin
        dat2:=$40; {clear SK, CLK}
        dat3:=$03; {set CS, R/W}
        port[B]:= dat2;
        port[C]:= dat3;
        dat2:= dat2 or $20; {set SK}
        port[B]:= dat2;
        ambil:= port[C];
        ambil:= ambil and $10;
        ambil:= ambil shr 4;
        ambil:= ambil shl (8 - J);
        simpan:= simpan or ambil;
    end;
    if simpan <> data[I] then goto gagal
end;
port[C]:= $02;      {clear FMC CS}
port[B]:= $40;      {clear SK, CLK sel}
port[B]:= $60;      {set SK}
port[B]:= $40;      {clear SK}
port[B]:= $60;      {set SK}
goto selesai;
gagal:
    delay(500);
    BuatTombol(0,530,370,80,35,2,0,15,7,'GAGAL' );
    goto done;
selesai:
    delay(500);
    BuatTombol(0,530,370,80,35,3,0,15,6,'O K !' );
done:
    repeat
        TombolMouseDitekan(Kanan,Tikus);
    until Tikus.TombolMana <> 0;
end;

```

```

procedure Running(kanal, output);
begin
    port[B]:=$0;      {reset FMC,clear step, CLK sel}
    port[B]:=$20;      {set step}
    port[B]:=$70;      {set rst, CLK sel}
    port[B]:=$50;      {clear step}
    port[B]:=$70;      {step step}
    for I:= 1 to 32 do
    begin
        port[B]:=$50; {clear step} {reset frame}
        port[B]:=$70; {step step}
    end;

```

```

Go:=$58 or kanal;
port[B]:=$50; {oscillator}
repeat
    port[B]:=Go;
    clear:=port[C];
    clear:=clear and $80;
    port[B]:=Go xor $08;
    until clear = 0;
    Outp:=port[$301]; {OE 574 buffer data output}
asm
    xor ah,ah
    mov outp,ax
end;
port[B]:=$40; {oscillator off}
end;

```

```

Procedure Hapus(warnalatar,x1,y1,x2,y2:integer);
begin
    setfillstyle(1{solidfill},warnalatar);
    bar(x1,y1,x2,y2);
end;

```

```

Function Kotak(x1,y1,x2,y2,rua1,rua2:integer):integer;
Var
    x,y,rua:integer;
Begin
    setcolor(rua1);
    line(x1,y1,x2,y1);
    line(x1,y1,x1,y2);
    setcolor(rua2);
    line(x2,y2,x2,y1);
    line(x2,y2,x1,y2);
End;

```

```

procedure bingkai;
begin
    kotak(1,1,638,478,black,white);
    kotak(2,2,638,478,black,white);
    kotak(2,2,638,31,black,white);
    kotak(9,37,209,245,white,black);
    kotak(12,40,206,242,black,white);
    kotak(12,40,206,67,black,white);
    kotak(220,37,420,245,white,black);
    kotak(223,40,417,242,black,white);
    kotak(223,40,417,67,black,white);
    kotak(434,40,628,67,black,white);
    kotak(431,37,631,245,white,black);
    kotak(434,40,628,242,black,white);
end;

```

```

Procedure Tampilan;
Begin
    Hapus(lightgray,0,0,getmaxx,getmaxy);
    hapus(blue,3,3,637,30);
    hapus(BLue,12,41,205,66);
    hapus(BLue,223,41,416,66);
    hapus(BLue,434,41,627,66);

```

```

hapus(black,60,85,190,230);
hapus(black,265,85,400,230);
hapus(black,475,85,610,230);

hapus(BLack,100,255,190,280);    {angka}
hapus(BLACK,310,255,400,280);
hapus(BLACK,520,255,610,280);
hapus(BLue,13,303,287,328);    {pengkondisian}
hapus(BLue,303,303,507,328);    {DATA}

{***** komentar *****}
setcolor(white);
settextstyle(gothicfont,0,1);
settextjustify(center,center,center);
Outtextxy(330,1,'Aplikasi Fuzzy Logic Control Pada Alat Quality
Control Air Minum PDAM');
settextjustify(left,center,center);
settextstyle(Smallfont,0,6);
setcolor(lightgreen);
Outtextxy(25,40,'Derajat keasaman');
Outtextxy(243,40,'Oksigen Terlarut');
Outtextxy(480,40,'Kejernihan ');
Outtextxy(385,305,'DATA');
settextstyle(Smallfont,0,5);
Outtextxy(30,307,'Proses Pengkondisian Air Minum');
Outtextxy(20,333,'Air Baku Minum');
Outtextxy(20,365,'Tawas');
Outtextxy(20,395,'K.Tohor');
Outtextxy(180,333,'Air Minum');
Outtextxy(165,365,'Tawas');
Outtextxy(165,395,'K.Tohor');

{***** Komentar di DATA *****}
settextstyle(Smallfont,0,5);
setcolor(red);
Outtextxy(313,330,'Air Baku Minum');
Outtextxy(313,398,'Air Minum');
setcolor(blue);
Outtextxy(313,357,'Awal');
Outtextxy(313,377,'Akhir');
Outtextxy(313,425,'Awal');
Outtextxy(313,445,'Akhir');

{penambahan cairan pengkondisi air minum}
BuatTombol(5,80,440,45,17,1,15,0,4,Pesan[7]);
BuatTombol(5,225,440,45,17,1,15,0,4,Pesan[7]);
BuatTombol(0,225,365,55,17,1,0,15,4,"");
BuatTombol(0,225,395,55,17,1,0,15,4,"");
BuatTombol(0,80,365,55,17,1,0,15,4,"");
BuatTombol(0,80,395,55,17,1,0,15,4,"");
str(VolMinum:2,X1);
BuatTombol(0,165,440,40,17,1,0,15,5,X1);
Outtextxy(200,445,'L');
str(VolBaku:2,X1);
BuatTombol(0,25,440,40,17,1,0,15,5,X1);
Outtextxy(60,445,'L');
Outtextxy(275,371,'L');    {penambahan tawas air minum}
Outtextxy(275,401,'L');    {penambahan tohor air minum}
Outtextxy(130,371,'L');    {penambahan tawas baku}
Outtextxy(130,401,'L');    {penambahan tohor baku}

{***** tempat DATA *****}
BuatTombol(0,355,426,45,17,1,0,15,4,"");
BuatTombol(0,405,426,45,17,1,0,15,4,"");
BuatTombol(0,455,426,45,17,1,0,15,4,"");
End;
procedure Simulasi(warna,X,Y,L,T : word; Tbl,cat,cat1,font:byte;
pesan:string);
begin
  BuatTombol(warna,X,Y,L,T,Tbl,cat,cat1,font,pesan);
end;

procedure BuatTombol(warna,X,Y,L,T : word; Tbl,cat,cat1,font:byte;
pesan:string);
begin
  hapus(warna,X,Y,X+L,Y+T);
  for I:= 1 to Tbl do
    begin
      kotak(X-I,Y-I,X+L+I,Y+T+I,cat,cat1);
    end;
    settextstyle(smallfont,0,font);
    setcolor(yellow);
    settextjustify(center,center,center);
    Outtextxy(X+L div 2,Y+T div 2-3,pesan);
  end;

function TombolDipilih(TTombol: PushButton; Cacah: byte): byte;
begin
  CursorMati;
  if MouseDidalamKotak(530,370,80,35) then TombolDipilih:=4;
  if MouseDidalamKotak(80,440,70,25) then TombolDipilih:=5;
  if MouseDidalamKotak(540,430,60,20) then TombolDipilih:=6;
  if MouseDidalamKotak(225,440,45,17) then TombolDipilih:=8;
end;

function pH:real;
begin
  Y:=0;
  port[A]:= 2;
  port[Start]:=0; {start konversi : low}

  DataADC:= port[Baca];
  DataADC:= (dataADC + 112.36) / 32.72;
  Str(DataADC:4,2,X1);
  hapus(BLack,100,255,189,279);    {angka}
  Outtextxy(110,259,X1);
  Y:= round(DataADC * 145 / 14);
  hapus(black,60,85,189,229);
  setcolor(magenta);
  line(60,229-Y,189,229-Y);
  delay(1);
  ph:=dataadc;
end;

function DOx:real;
begin
  Y:=0;
  port[A]:= 4;
  port[Start]:=0; {start konversi : low}

  DataADC:= port[Baca];
  DataADC:= (DataADC + 22.9) / 8.1;
  Str(DataADC:4,2,X1);
  hapus(BLACK,310,255,399,279);
  Outtextxy(320,259,X1);
  Y:= round(DataADC * 145 / 20);
  hapus(black,265,85,399,229);
  setcolor(magenta);
  line(265,229-Y,399,229-Y);
  delay(1);
  Dox:=DataADC;
end;

function Jernih:real;
begin
  Y:=0;
  port[A]:= $00;
  port[Start]:=0; {start konversi : low}

  DataADC:= port[Baca];
  DataADC:= DataADC * 6 / 100;
  NTU:=(DataADC-3.82)/(-0.18);
  Str(NTU:4,2,X1);

```



```

port[A]:=$78;    {pengaduk 2}
delay(17000);
port[A]:=0;

port[A]:=$78;
delay(3000);
port[A]:=0
end;

procedure Aerator;
begin
port[A]:=$38;    {Aerasi}
port[A]:=$78;
delay(10000);
port[A]:=0;
end;

procedure TohorBanyak;
begin
port[A]:=$78; {pengaduk 1}
delay(50);
port[A]:=$58; {katup 3}
delay(120000);
port[A]:=0;

port[A]:=$78;
delay(5000);
port[A]:=0;
end;

procedure Tohor;
begin
port[A]:=$58;    {katup 3}
delay(50);
port[A]:=$78;    {pengaduk 1}
delay(60000);
port[A]:=0;

delay(50);
port[A]:=$78;
delay(5000);
port[A]:=0;
end;

procedure TawasBanyak;
begin
port[A]:=$98;    {katup 2}
delay(50);
port[A]:=$78;    {pengaduk 1}
delay(60000);
port[A]:=0;

delay(50);
port[A]:=$78;
delay(3000);
port[A]:=0;
end;

procedure Tawas;
begin
port[A]:=$98;    {katup 2}
delay(50);
port[A]:=$78;    {pengaduk 1}
delay(30000);
port[A]:=0;

delay(50);
port[A]:=$78;
delay(5000);
port[A]:=0;
end;

```

```

procedure DOTawas05;
begin
port[A]:=$98;    {katup 2}
delay(50);
port[A]:=$78;    {pengaduk 2}
delay(50);
port[A]:=$38;    {pump}
delay(17000);
port[A]:=0;

delay(50);
port[A]:=$78;
delay(5000);
port[A]:=0
end;

procedure DOTohorBanyak;
begin
port[A]:=$58;    {katup3}
delay(50);
port[A]:=$78;    {pengaduk 1}
delay(50);
port[A]:=$38;    {pump}
delay(120000);
port[A]:=0;

delay(50);
port[A]:=$78;
delay(3000);
port[A]:=0;
end;

procedure DOTohor;
begin
port[A]:=$58;    {katup3}
delay(50);
port[A]:=$78;    {pengaduk 1}
delay(50);
port[A]:=$38;    {pump}
delay(60000);
port[A]:=0;

delay(50);
port[A]:=$78;
delay(3000);
port[A]:=0;
end;

procedure DOTawasBanyak;
begin
port[A]:=$98;    {Katup2}
delay(50);
port[A]:=$78;    {pengaduk 1}
delay(50);
port[A]:=$38;    {pump}
delay(60000);
port[A]:=0;

delay(50);
port[A]:=$78;
delay(3000);
port[A]:=0;
end;

procedure DOTawas;
begin
port[A]:=$98;    {Katup2}
delay(50);
port[A]:=$78;    {pengaduk 1}
delay(50);
port[A]:=$38;    {pump}

```

```

delay(30000);
port[A]:=0;

delay(50);
port[A]:=$78;
delay(3000);
port[A]:=0;
end;

*****
**      AIR MINUM      **
*****

procedure proses1(Var AddTawas1,AddTohor1:real);
var
  FuzOutpH,FuzOutDOx,FuzOutJernih : integer;
  TbhTawas1,TbhTohor1 : real;
begin
  pH;
  FuzOutpH:=running(0,pH);
  DOx;
  FuzOutDOx:=running(1,DOx);
  Jernih;
  FuzOutJernih:=running(Jernih);

Defuzz1(FuzOutpH,FuzOutDOx,FuzOutJernih,TbhTawas1,TbhTohor1);

  pH;DOx;Jernih;
  delay(100);
end;
AddTawas1:=TbhTawas1+AddTawas1;
AddTohor1:=TbhTohor1+AddTohor1;
end;

procedure Defuzz1(FuzOutpH,FuzOutDOx,FuzOutJernih:integer;
  var TbhTawas1,TbhTohor1 : real);
begin
  TbhTawas1:=0;
  TbhTohor1:=0;
  if (FuzOutJernih < 70) and (FuzOutpH <= 25)then
  begin
    sound(500);delay(100); {buang}
    buatTombol(8,530,300,70,25,3,15,0,5,'Buang');
    TbhTawas1:=TbhTawas1+0;
    TbhTohor1:=TbhTohor1+0;
    nosound;
    exit;
  end;

  if (FuzOutpH = 50) and (FuzOutJernih = 70) and (FuzOutDOx = 50)
  then
  begin
    sound(200);
    bubar:=0;
    buatTombol(8,530,300,70,25,3,15,0,5,'Optimum');
    delay(100);
    TbhTawas1:=TbhTawas1+0;
    TbhTohor1:=TbhTohor1+0;
    nosound;
    exit;
  end;

  case FuzOutDOx of
    50 : begin
      if (FuzOutpH = 0) and (FuzOutJernih >= 70) then
      begin
        TohorBanyak1;
        TbhTohor1:=TbhTohor1+4.1;
        TbhTawas1:=TbhTawas1+0;exit;
      end;
      if (FuzOutpH = 50) and (FuzOutJernih < 70 ) then
      begin
        Tawas15;
        TbhTohor1:=TbhTohor1+0;
        TbhTawas1:=TbhTawas1+0.5;exit;
      end;
      if (FuzOutpH = 25) and (FuzOutJernih >= 70) then
      begin
        Tohor1;
        TbhTohor1:=TbhTohor1+2.1;
        TbhTawas1:=TbhTawas1+0;exit;
      end;
      if (FuzOutpH = 75) and (FuzOutJernih >= 70) then
      begin
        Tawas1;
        TbhTawas1:=TbhTawas1+1.1;
        TbhTohor1:=TbhTohor1+0;exit;
      end;
      if (FuzOutpH = 100) and (FuzOutJernih >= 70) then
      begin
        TohorBanyak1;
        TbhTawas1:=TbhTawas1+2.1;
        TbhTohor1:=TbhTohor1+0;exit;
      end;
    end;
  0 : begin
    if (FuzOutpH = 0) and (FuzOutJernih >= 70) then
    begin
      DOTohorBanyak1;
      TbhTohor1:=TbhTohor1+4.1;
      TbhTawas1:=TbhTawas1+0;exit;
    end;
    if (FuzOutpH = 50) and (FuzOutJernih < 70 ) then
    begin
      DOTawas15;
      TbhTohor1:=TbhTohor1+0;
      TbhTawas1:=TbhTawas1+0.5;exit;
    end;
    if (FuzOutpH = 25) and (FuzOutJernih >= 70) then
    begin
      DOTohor1;
      TbhTohor1:=TbhTohor1+2.1;
      TbhTawas1:=TbhTawas1+0;exit;
    end;
    if (FuzOutpH = 50) and (FuzOutJernih >= 70) then
    begin
      Aerator1;
      TbhTohor1:=TbhTohor1+0;
      TbhTawas1:=TbhTawas1+0;exit;
    end;
    if (FuzOutpH = 75) and (FuzOutJernih >= 70) then
    begin
      DOTawas1;
      TbhTohor1:=TbhTohor1+0;
      TbhTawas1:=TbhTawas1+1.1;exit;
    end;
    if (FuzOutpH = 100) and (FuzOutJernih >= 70) then
    begin
      DOTawasBanyak1;
      TbhTohor1:=TbhTohor1+0;
      TbhTawas1:=TbhTawas1+2.1;exit;
    end;
  end;
  end;
  buatTombol(8,530,300,70,25,3,15,0,5,'Buang');
end;

procedure Tawas15;
begin
  port[A]:=$d8; {katup 4}
  delay(50);
  port[A]:=$f8; {pengaduk 2}
  delay(15000);

```



```

port[A]:=0;
delay(50);
port[A]:=$f8;
delay(3000);
port[A]:=0
end;

procedure Aerator1;
begin
port[A]:=$38; {Aerasi}
delay(50);
port[A]:=$f8;
delay(10000);
port[A]:=0;
end;

procedure TohorBanyak1;
begin
port[A]:=$18; {katup 1}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(120000);
port[A]:=0;

delay(50);
port[A]:=$f8; {pengaduk 2}
delay(3000);
port[A]:=0;
end;

procedure Tohor1;
begin
port[A]:=$18; {katup 1}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(60000);
port[A]:=0;

delay(50);
port[A]:=$f8; {pengaduk 2}
delay(3000);
port[A]:=0;
end;

procedure TawasBanyak1;
begin
port[A]:=$d8; {katup 4}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(60000);
port[A]:=0;

delay(50);
port[A]:=$f8; {pengaduk 2}
delay(3000);
port[A]:=0;
end;

procedure Tawas1;
begin
port[A]:=$d8; {katup 4}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(30000);
port[A]:=0;

delay(50);
port[A]:=$f8; {pengaduk 2}
delay(3000);
port[A]:=0;
end;

procedure DOTawas15;
begin
port[A]:=$d8; {katup 4}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(17000);
port[A]:=0;

delay(50);
port[A]:=$f8;
delay(3000);
port[A]:=0
end;

procedure DOTohorBanyak1;
begin
port[A]:=$18; {katup 1}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(50);
port[A]:=$38; {pump}
delay(120000);
port[A]:=0;

delay(50);
port[A]:=$f8;
delay(3000);
port[A]:=0;
end;

procedure DOTohor1;
begin
port[A]:=$18; {katup 1}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(50);
port[A]:=$38; {pump}
delay(60000);
port[A]:=0;

delay(50);
port[A]:=$f8;
delay(3000);
port[A]:=0;
end;

procedure DOTawasBanyak1;
begin
port[A]:=$d8; {Katup 4}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(50);
port[A]:=$38; {pump}
delay(60000);
port[A]:=0;

delay(50);
port[A]:=$f8;
delay(3000);
port[A]:=0;
end;

procedure DOTawas1;
begin
port[A]:=$18; {Katup 4}
delay(50);
port[A]:=$f8; {pengaduk 2}
delay(50);
port[A]:=$38; {pump}
delay(30000);
port[A]:=0;
end;

```



```

delay(50);
port[A]:=$f8;
delay(3000);
port[A]:=0;
end;

{-----Program Utama-----}
Begin
  AddTawas :=0;
  AddTawas1:=0;
  AddTohor :=0;
  AddTohor1:=0;
  clrscr;

  gotoxy(10,10);write("Volume Air Baku : liter");
  gotoxy(30,10);readln(VolBaku);
  gotoxy(10,12);write("Volume Air Minum : liter");
  gotoxy(30,12);readln(VolMinum);
  Kode:=DETECT;
  Initgraph(kode,mode,'C:\TP\BGI');
  InitMouse(Stat,CTombol);

  port[CW]:=$88;
  port[A]:=$00;
  Tampilan;

  if stat then
  begin
    ubahBentukKursor(tangan);
    kursorHidup;
  end;

  repeat
  repeat
    {menunggu tombol kiri mouse diklik}
    KursorHidup;
  repeat
    kursormati;
    ph;
    kursorhidup;
    dox;
    kursormati;
    jernih;
    kursorhidup;
    TombolMouseDitekan(Kiri,Tikus);
  until Tikus.TombolMana<>0;
  sound(150);delay(100);nosound;
  BoxDipilih:= TombolDipilih(Petak,CacahT);
  until BoxDipilih <> 0;

  case BoxDipilih of
  4 : begin
    Simulasi(0,530,370,80,35,3,0,15,5,Pesan[4] );
    Download;
    BuatTombol(9,530,370,80,35,3,15,0,5,Pesan[4] );
    KursorHidup;
  end;
  5 : begin
    Simulasi(0,80,440,45,17,1,0,15,4,Pesan[7] ); {air baku}
    x:=ph;
    str(x:4:2,x1);
    setcolor(yellow);
    hapus(black,355,359,399,375); {hapus angka awal pH}
    outtextxy(365,359,X1);
    X:=DOx;
    str(x:4:2,X2);
    setcolor(yellow);
    hapus(black,405,359,449,375); {hapus angka awal DO}
    outtextxy(413,359,X2);
    X:=Jernih;

    str(X:4:2,X3);
    setcolor(yellow);
    hapus(black,454,359,499,375); {hapus angka awal Jernih}
    outtextxy(460,359,X3);

    proses(AddTawas,AddTohor);

    setcolor(yellow);
    hapus(BLack,80,365,125,380); {hapus angka penambahan
                                     tawas}

    x:=AddTawas;
    str(x:4:1,x1);
    Outtextxy(95,365,x1); {angka penambahan tawas
                             baku}
    hapus(BLack,80,395,125,410); {hapus angka penambahan
                                     tohor}

    x:=AddTohor;
    str(x:4:1,x2);
    Outtextxy(95,395,x2); {angka penambahan tohor baku}
    BuatTombol(5,80,440,45,17,1,15,0,4,Pesan[7] );
    hapus(lightgray,527,297,603,328);
    KursorHidup;nosound;
  end;
  8 : begin
    BuatTombol(0,225,440,45,17,1,0,15,4,Pesan[7] );
    x:=ph;
    str(x:4:2,x1);
    setcolor(yellow); { AIR MINUM }
    hapus(black,355,426,399,442); {hapus angka awal pH}
    outtextxy(365,426,X1);

    X:=DOx;
    str(x:4:2,X2);
    setcolor(yellow);
    hapus(black,404,426,449,442); {hapus angka awal DO}
    outtextxy(413,426,X2);

    X:=Jernih;
    str(X:4:2,X3);
    setcolor(yellow);
    hapus(black,454,426,499,442); {hapus angka awal Jernih}
    outtextxy(460,426,X3);

    proses1(AddTawas1,AddTohor1);

    setcolor(yellow);
    hapus(BLack,225,365,270,380); {hapus angka penambahan
                                     tawas}

    x:=AddTawas1;
    str(x:4:1,x1);
    Outtextxy(240,365,x1); {angka penambahan tawas}
    hapus(BLack,225,395,270,410); {hapus angka penambahan
                                     tohor}

    x:=AddTohor1;
    str(x:4:1,x2);
    Outtextxy(240,395,x2); {angka penambahan tohor}
    nosound;
    BuatTombol(5,225,440,45,17,1,15,0,4,Pesan[7] );
    KursorHidup;
  end;
  6 : begin
    Simulasi(0,540,430,60,20,2,0,15,5,Pesan[6] );
    delay(300);
    BuatTombol(13,540,430,60,20,2,15,0,5,Pesan[6] );
    delay(100);
    exit;
  end;
  end;
  Until BoxDipilih = 9;
  Closegraph;
end.

```



Adaptive Logic

Data Sheet

NLX230

FUZZY MICROCONTROLLER™

The Fuzzy MicroController (FMC), Model NLX230, is a fully configurable VLSI Fuzzy Logic engine. It is one in a family of MicroController devices offered by Neuralogix. The FMC devices are intended to augment or supplant conventional microprocessor implementation in performance or cost-critical embedded control systems.

Instead of using algorithms executed sequentially to control an output based on input conditions, as is done in general purpose microprocessor implementations, the devices employ the principles of Fuzzy Logic to calculate an optimal output action based on input conditions, thereby performing a parallel operation to control the output. This efficient implementation allows high processing rates (30 million rules/second) at low cost.

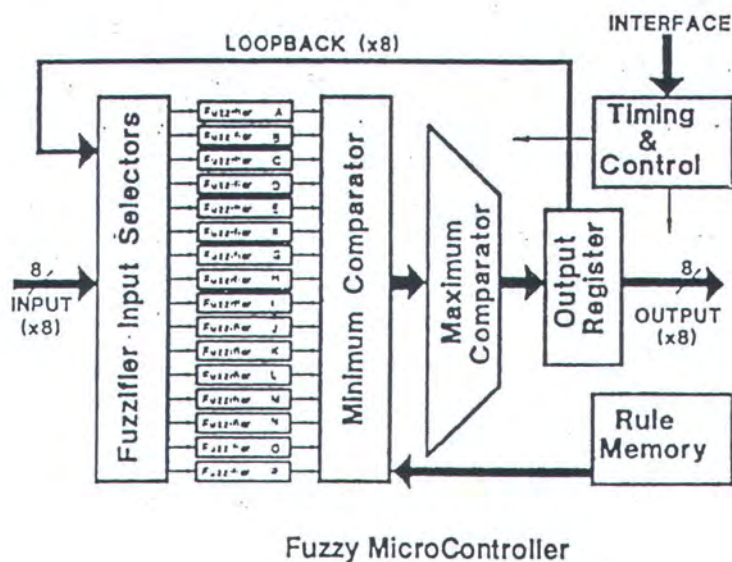
Patent Pending

Applications

- ☐ Enhance Conventional PID Controllers
- ☐ Smart Appliances
- ☐ Pattern Matching
- ☐ Sequencers, State Machines and Timers
- ☐ Automotive Applications
- ☐ Robotics
- ☐ Approximate Reasoning
- ☐ Expert Systems

Features

- ☐ Powerful Fuzzy Logic Processing (30M Rules/Sec)
- ☐ Simple, Low Cost PC Based Development Environment
- ☐ Cascadable
- ☐ Easy to Configure
- ☐ Minimal External Components
- ☐ Alternate Packaging Available
- ☐ Low Power CMOS
- ☐ Serial EEPROM or Micro Interface



| | | | |
|------|----|----|-----|
| RSTn | 1 | 40 | VDD |
| DIO | 2 | 39 | DO0 |
| DI1 | 3 | 38 | DO1 |
| DI2 | 4 | 37 | DO2 |
| DI3 | 5 | 36 | DO3 |
| DI4 | 6 | 35 | DO4 |
| DI5 | 7 | 34 | DO5 |
| DI6 | 8 | 33 | DO6 |
| DI7 | 9 | 32 | DO7 |
| VSS | 10 | 31 | VSS |
| SK | 11 | 30 | MA0 |
| CS | 12 | 29 | MA1 |
| DI | 13 | 28 | MA2 |
| DO | 14 | 27 | STB |
| M/Sn | 15 | 26 | CLK |
| R/Wn | 16 | 25 | XO |
| NC | 17 | 24 | XI |
| NC | 18 | 23 | NC |
| NC | 19 | 22 | NC |
| VSS | 20 | 21 | NC |

Standard 40-Pin
Connection Diagram

CHARACTERISTICS

| Absolute Maximum Ratings ($T_A = +25^{\circ}\text{C}$) | | | | | |
|--|-----------------------|-------|--------------|--------------------|------------------------|
| Symbol | Parameter | Min | Max | Units | Notes |
| V_{DO} | DC Supply Voltage | -0.3 | 7.0 | V | |
| V_{IN} | Input Pin Voltage | 0.3 | $V_{DO}+0.3$ | V | |
| I_{IN} | Input Pin Current | -10.0 | 10.0 | μA | @ 25°C |
| T_{OPR} | Operating Temperature | 0 | 70 | $^{\circ}\text{C}$ | |
| T_{STRG} | Storage Temperature | -55 | 150 | $^{\circ}\text{C}$ | |
| T_{LEAD} | Lead Temperature | — | 300 | $^{\circ}\text{C}$ | 10 sec |

| DC CHARACTERISTICS ($V_{DO}=5V\pm5\%$; $T_A=0$ to $+70^{\circ}\text{C}$) | | | | | |
|---|--------------------------------------|------|----------|---------------|------------------------------|
| Symbol | Parameter | Min | Max | Units | Notes |
| V_{IH} | Input High Voltage | 2.0 | V_{DO} | V | |
| V_{IL} | Input Low Voltage | -0.3 | +0.8 | V | |
| I_{IN} | Input Leakage Current | — | 5.0 | nA | @ 70°C |
| I_{IN} | Input Leakage Current Bi-Directional | — | 40.0 | nA | @ 70°C |
| I_{IH} | Input High Current | -1.0 | — | μA | $V_{IN}=5.5$ $V_{DO}=5.5$ |
| I_{IL} | Input Low Current | 1.0 | — | μA | $V_{IN}=0.0$ $V_{DO}=5.5$ |
| V_{OH} | Output High Voltage | 2.4 | — | V | $V_{DO}=5.0$ |
| V_{OL} | Output Low Voltage | — | 0.4 | V | $V_{DO}=5.0$ |
| I_{OH} | Output High Current | -8.0 | — | mA | $V_{OH}=2.4$ |
| I_{OL} | Output Low Current | 8.0 | — | mA | $V_{OH}=0.4$ |
| C_{IN} | Input Capacitance | — | 5 | pF | f=1 MHz |
| C_{IN} | Input Cap. Bi-Directional | — | 15 | pF | f=1 MHz |
| C_{OUT} | Output Capacitance | — | 10 | pF | f=1 MHz |
| P_D | Power Dissipation | .001 | 300 | mW | f=0 MHz f=30 MHz |

PIN DEFINITIONS

| SIGNAL NAME | PIN NUMBERS | SIGNAL | DIR. |
|--------------------------|------------------------------------|--|------|
| Data Input/Output | | | |
| DI0 | 2 | Data Input, bit 0 | In |
| DI1 | 3 | Data Input, bit 1 | |
| DI2 | 4 | Data Input, bit 2 | |
| DI3 | 5 | Data Input, bit 3 | |
| DI4 | 6 | Data Input, bit 4 | |
| DI5 | 7 | Data Input, bit 5 | |
| DI6 | 8 | Data Input, bit 6 | |
| DI7 | 9 | Data Input, bit 7 | |
| DO0 | 39 | Data Output, bit 0 | Out |
| DO1 | 38 | Data Output, bit 1 | |
| DO2 | 37 | Data Output, bit 2 | |
| DO3 | 36 | Data Output, bit 3 | |
| DO4 | 35 | Data Output, bit 4 | |
| DO5 | 34 | Data Output, bit 5 | |
| DO6 | 33 | Data Output, bit 6 | |
| DO7 | 32 | Data Output, bit 7 | |
| Data Control | | | |
| STB | 27 | Strobe - Used in conjunction with MA0-MA2 to strobe Input/Output data | Out |
| MA0 | 30 | Input/Output Mux Address bit 0 output | Out |
| MA1 | 29 | Input/Output Mux Address bit 1 output | Out |
| MA2 | 28 | Input/Output Mux Address bit 2 output | Out |
| EEPROM Interface | | | |
| SK | 11 | Serial Clock, Used to clock configuration data into device (Input when M/S = 0, Output when M/S = 1) | In |
| CS | 12 | Chip Select, Used in conjunction with M/S, and R/W to enable Serial data Input/Output, (Output when M/S = 1, Input when M/S = 0) | In |
| DI | 13 | Serial Data Input | In |
| DO | 14 | Serial Data Output | Out |
| R/Wn | 16 | Read/Write Input pin, used in Slave mode only. (Read = 1, Write = 0) | In |
| M/Sn | 15 | Master/Slave Configuration pin (Master when M/S = 1, Slave when M/S = 0) | In |
| Common Control | | | |
| XI | 24 | Crystal Oscillator Input | In |
| XO | 25 | Crystal Oscillator Output | Out |
| CLK | 26 | Buffered System Clock Output | Out |
| RSTn | 1 | Reset Input, low active | In |
| Power Inputs | | | |
| NC | 10, 17, 18, 19, 20, 21, 22, 23, 31 | Ground | |
| VDD | 40 | +5 volts | |

PIN DESCRIPTIONS

| PIN | DESCRIPTION | PIN | DESCRIPTION |
|---------|--|------|--|
| DI(7:0) | Data Input. Data is Time Division Multiplexed (TDM) as eight 8-bit bytes based on the decode of MA(2:0) and STB. | M/Sn | Master/Slave mode Control. Part of the configuration interface. In the master mode (M/Sn=logic 1), the device outputs the necessary control signals to read the EEPROM immediately following the negation of the RST input (transition of logic 0 to logic 1). The SK, CS and DO signals become active outputs, issuing the data sequences to read a 93C56 EEPROM via the DI input. In the slave mode, the device waits for the host system to download via the SK, CS, R/Wn and DI inputs, or to be read via the SK, CS, R/Nn inputs and the DO inputs. |
| DO(7:0) | Data Output. Data is TDM as eight 8-bit bytes based on the decode of MA(2:0) and STB. | XI | Crystal Oscillator Input. Utilized in conjunction with the XO output as part of the on-chip oscillator. This pin may also be directly driven by TTL compatible circuitry to input an external clock. |
| STB | Strobe. Used in conjunction with MA(2:0) to decode the input and output channel which is currently active in the device. | XO | Crystal Oscillator Output. Utilized in conjunction with the XI input as part of the on-chip oscillator. If an external clock is utilized, this output pin should be left unconnected. |
| MA(2:0) | MUX Address. Three-bit output which decodes the current input or output channel which is currently active in the device. | CLK | Clock Input. The buffered output of the internal oscillator for use as an off-chip clock source. |
| SK | Serial Clock. Part of the configuration interface. In the master mode, SK is utilized to provide the clock output to the external EEPROM. In this mode of operation, the SK frequency is the input clock divided by 32 (XI/32). In the slave mode, SK is utilized as the clock input provided by the host system during download. | RSTn | Reset. Chip reset input (active logic 0). When asserted logic 0 for one or more clock transitions (low to high), the device halts. In master mode (M/Sn = logic 1), upon the negation of RSTn, the device outputs the necessary configuration interface signaling to download the internal memory. Upon completion or the download, the device begins processing input data. In the slave mode (M/Sn = logic 0), upon the negation of RSTn, the device begins processing input data. (Note, the device requires configuration from a host system prior to obtaining predictable results.) |
| CSn | Chip Select. Part of the configuration interface. In the master mode, CS is utilized to provide the chip select output to the external EEPROM. In the slave mode, CS is utilized to input (active low) the chip select function of an external address decoder from the host system during download. | | |
| DI | Serial Data Input. Part of the configuration interface. In the master mode, DI is utilized to input serial data from the external EEPROM's data output pin. In the slave mode, DI is utilized to input the serial data from the host system. | | |
| DO | Serial Data Output. Part of the configuration interface. In the master mode, DO is utilized to provide the necessary control function (read request) and the address to the external EEPROM's data input. In the slave mode, DO is utilized to output the contents of the internal configuration memory to the host system. | | |
| R/Wn | Read/Write. Part of the configuration interface. In the master mode, this input has no function, and should be terminated with an external pull-up resistor. In the slave mode, R/Wn is utilized to provide the read request of the contents of the internal configuration memory (R/Wn = logic 1), or to provide the write command to the device (R/Wn = logic 0). | | |

INTERNAL REGISTERS

| | | | | | | | |
|--|-------------|-------------|-------------|-------------|-------------|------------|------------|
| OUTPUT MODE REGISTER Address Offset 0C _{hex} (Bits 96 thru 103) | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Out 7 | Out 6 | Out 5 | Out 4 | Out 3 | Out 2 | Out 1 | Out 0 |
| 0 = Initial Value + Action 1 = Previous Output + Action | | | | | | | |
| INPUT CONFIGURATION REGISTER Address Offset 0D _{hex} (Bits 104 thru 111) | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| In 7 | In 6 | In 5 | In 4 | In 3 | In 2 | In 1 | In 0 |
| 0 = Feedback Output 1 = External Input | | | | | | | |
| MEMBERSHIP FUNCTION TYPE REGISTER 0 & 1 Address Offset 0E _{hex} (Bits 112 thru 127) | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Fuzzy 7 | Fuzzy 6 | Fuzzy 5 | Fuzzy 4 | Fuzzy 3 | Fuzzy 2 | Fuzzy 1 | Fuzzy 0 |
| Fuzzy 15 | Fuzzy 14 | Fuzzy 13 | Fuzzy 12 | Fuzzy 11 | Fuzzy 10 | Fuzzy 9 | Fuzzy 8 |
| 0 = Inclusive Membership 1 = Exclusive Membership | | | | | | | |
| INITIAL VALUE REGISTERS 0 THRU 7 Address Offset 10 _{hex} thru 17 _{hex} (Bits 128 thru 191) | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Sign | MSB | | | | | | LSB |
| 8-Bit signed initial value for each of the eight outputs. | | | | | | | |
| RULE BOUNDARY REGISTERS 0 THRU 7 Address Offset 18 _{hex} thru 1F _{hex} (Bits 192 thru 255) | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Logic 0 | Logic 0 | MSB | | | | | LSB |
| 6-Bit Rule Number of the last rule associated with like-numbered output. | | | | | | | |

The Output Mode Register determines whether each of the eight outputs will be in the Immediate or Accumulate mode. When an associated bit is set to Logic 0, the output is set equal to the sum of the Initial Value plus the Action Value. When an associated bit is set to Logic 1, the output is set equal to the sum of the Previous Output plus the Action Value.

The device's input configuration information is contained in this register. Input to a Fuzzifier can be from external Crisp Inputs or from internal Loop Back from the Outputs. When the associated I/O register bit is set to Logic 1, Input is applied data from the Input Pins when the associated I/O Register Bit is set to Logic 0, Inputs are applied from the Loop Back Outputs.

Each Fuzzifier has associated with it the option of complementing its function. Each register provides control for eight Fuzzifiers. When a bit is set to Logic 0, the function is Inclusive. When a bit is set to Logic 1, the function is Exclusive.

Each output has an Initial Value register associated with it. Each register contains a user defined Initial Value (range -128 to +127), which corresponds, in the Immediate mode, to the nominal output value for that output with no corrections. In the Accumulate mode the Initial Value is used to calculate the first output after reset.

Each output from the device uses a certain number of contiguous rules. The number of rules used by each output is equal to the difference of the corresponding output rule boundary register and the preceding output rule boundary register. Each register contains the address of the last rule used by the corresponding output. Each output used must have at least two rules assigned to it. Note that rule 0 is not user editable, and should be treated as an empty rule associated with output 0.

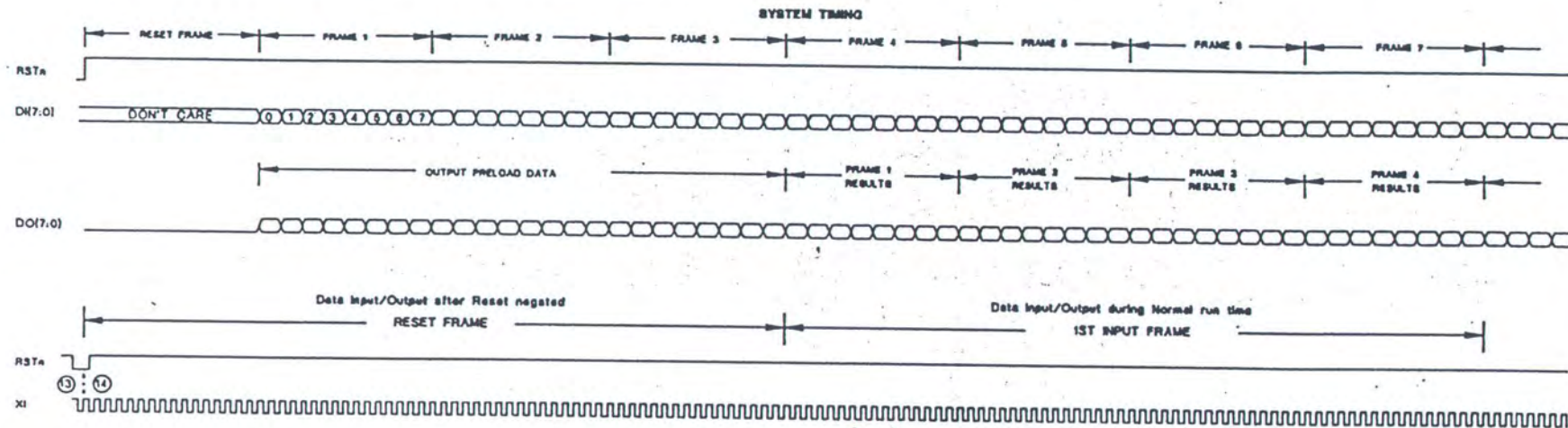
INTERNAL REGISTERS

| TERM REGISTERS 0 THRU 15 Address Offset 20 _{hex} thru 3F _{hex} (Bits 256 thru 511) | | | | | | | |
|--|-------------|-------------|--------------|-------------|-------------|------------|---------------|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Center MSB | | | | | | | Center LSB |
| Select 2 | Select 1 | Select 0 | Width MSB | | | | Width LSB |
| | | | | | | | |
| | | | | | | | |
| ACTION VALUE REGISTERS 0 THRU 63 Address Offset 40 _{hex} thru 7F _{hex} (Bits 512 thru 1023) | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Sign | MSB | | | | | | LSB |
| Two's Complement action values for rules 0 thru 63 | | | | | | | |
| RULE REGISTERS 0 THRU 63 Address Offset 80 _{hex} thru BF _{hex} (Bits 1024 thru 1535) | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Fuzzy 7 | Fuzzy 6 | Fuzzy 5 | Fuzzy 4 | Fuzzy 3 | Fuzzy 2 | Fuzzy 1 | Fuzzy 0 |
| Address Offset C0 _{hex} thru FF _{hex} (Bits 1536 thru 2047) | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Fuzzy 15 | Fuzzy 14 | Fuzzy 13 | Fuzzy 12 | Fuzzy 11 | Fuzzy 10 | Fuzzy 9 | Fuzzy 8 |
| 0 = Exclude term 1 = Include term | | | | | | | |

Term Registers contain the Center Locations, Input Selections and Width values. Each Fuzzifier has a 8-bit Center Location associated with it. The value in the Center Location represents the input value with the highest possible degree of membership for that particular true membership function. The second address contains the Input Selections and the Width values. Each Fuzzifier has a 3-bit Input Select Register which allows the user to program which input (or output Loop Back) the Fuzzifier will use as an input data source. The most significant three bits, binary value (0-7), selects an input (1-8). The least significant 5 bits represent the Width of the membership function; i.e., the maximum distance from the specified Center Location which has any degree of membership at all. Distances outside of this range will result in a degree of membership of zero (0), for Inclusive functions.

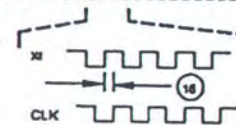
Each Rule has associated with it a twos complement Action Value. The most significant bit contains the sign. The following seven bits contain the Action Value (range -128 to +127). The Action Values pertain only to the output with which this rule is associated.

Each Rule is made up of a selection of the Fuzzifiers. The rule Register selects which Fuzzifiers are included in the rule. Address offsets 80_{hex} thru BF_{hex} select from the lower eight Fuzzifiers for rules 0 thru 63. Address offsets C0_{hex} thru FF_{hex} select from the upper eight Fuzzifiers for rules 0 thru 63. If the bit is logic 1, the corresponding Fuzzifier is included. Otherwise, the Fuzzifier is not included. For internal timing reasons, rule 0 is not user editable, so addresses 80_{hex} and C0_{hex} should be written to logic 0.

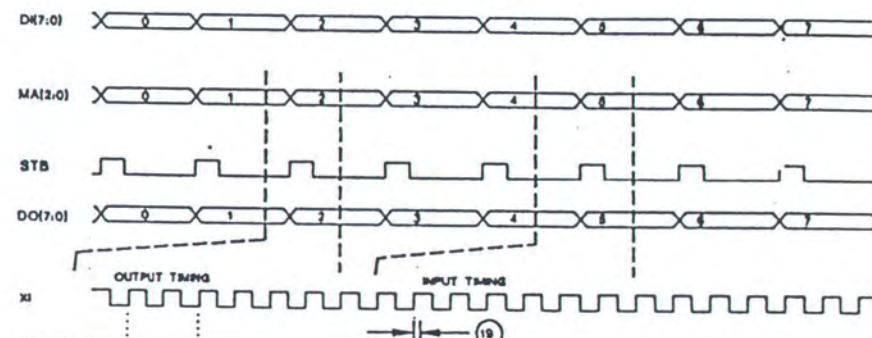


LEGEND

| ITEM | SYMBOL | DESCRIPTION | Min | Typ | Max | Unit | NOTES |
|----------------|------------------|-------------------------------|-----|------|-----|------|-------|
| 1 | T _{CK} | CLOCK PERIOD (X1) | | | 30 | nHz | |
| 2 | T _{SK} | CLOCK PERIOD, SK | | | | nHz | XV32 |
| 3 | T _{CSK} | DELAY FROM XI RISING TO SK | | | 24 | nS | |
| 4 | T _{CS} | SK RISING TO CS | | | 32 | nS | |
| 5 | T _{DO} | SK RISING TO DO | | | 18 | nS | |
| 6 | T _{DS} | DI SETUP TO SK RISING | | 5 | | nS | |
| 7 | T _{DH} | DI HOLD | 5 | | | nS | |
| 8 | T _{SK} | CLOCK PERIOD, SLAVE MODE (SK) | | 1000 | | nS | |
| 9 | T _{CS} | SETUP CS TO SK | 5 | | | nS | |
| 10 | T _{DS} | SETUP DI TO SK | 5 | | | nS | |
| 11 | T _{DO} | SETUP DO TO SK | 5 | | | nS | |
| 12 | T _{PD} | PROPAGATION SK TO DO | | | 18 | nS | |
| DATA INTERFACE | | | | | | | |
| 13 | T _{SR} | SETUP RSTn TO XI | 5 | | | nS | |
| 14 | T _{HR} | HOLD RSTn TO XI | 5 | | | nS | |
| 15 | T _{PC} | PROPAGATION XI TO CLK | | | 10 | nS | |
| 16 | T _{SI} | SETUP DI(0-7) TO XI | 5 | | | nS | |

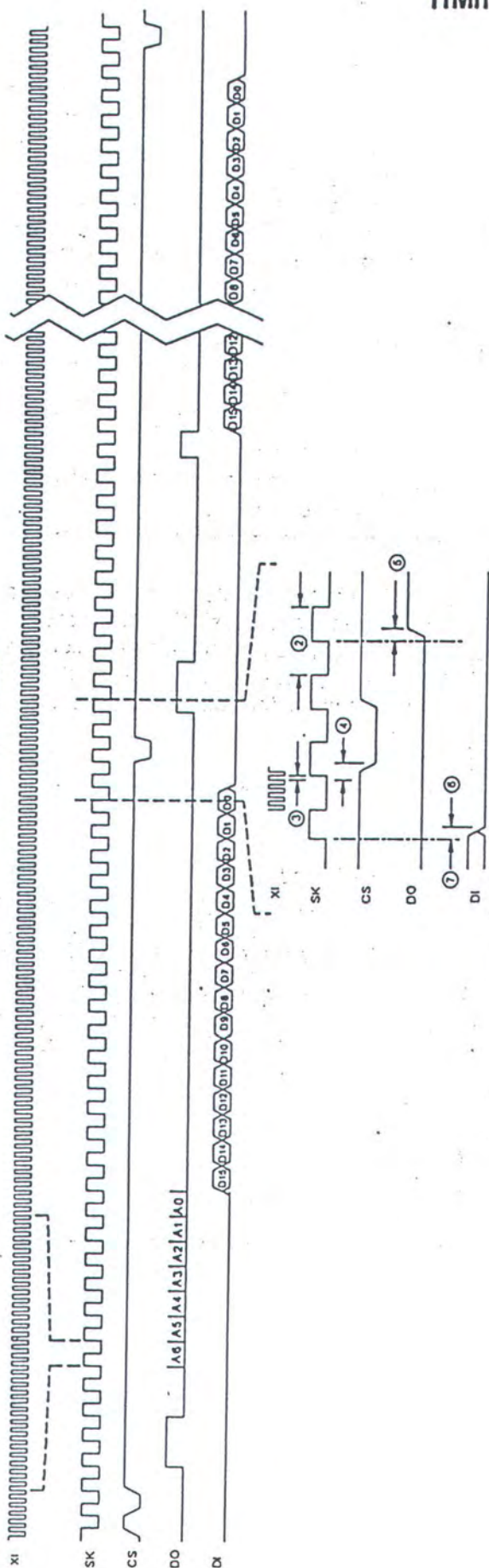


TYPICAL FRAME

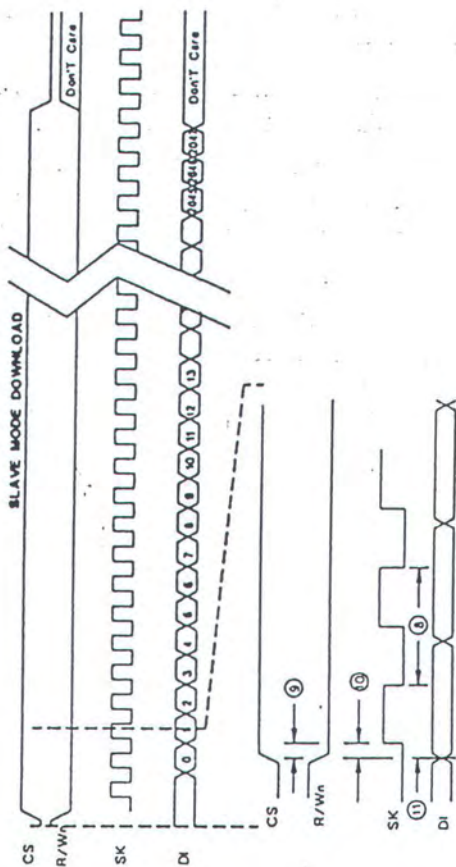


TIMING

MASTER MODE DOWNLOAD



SLAVE MODE DOWNLOAD



SLAVE MODE UPLOAD

